

# IBM 2835 Storage Control and IBM 2305 Fixed Head Storage Module

## Reference Manual

Cross-System

**IBM**



---

# **IBM 2835 Storage Control and IBM 2305 Fixed Head Storage Module**

---

## **Reference Manual**

---

---

## **Cross-System**

---

---

Publication Number  
GA26-1589-5

File Number  
S/360-07, S/370-07, 4300-07

---

## **Federal Communications Commission (FCC) Statement**

**Warning:** This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. As temporarily permitted by regulation it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area is likely to cause interference in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

In addition to the FCC statement above, the user of this manual should be aware that the referenced statement applies only to devices manufactured after January 1, 1981, and used in the United States.

## **Sixth Edition (October 1983)**

This publication is a major revision that replaces and makes Order No. GA26-1589-4 obsolete. All changes and additions contained in Technical Newsletter GN26-0360 have been incorporated into this edition. Periodically changes or additions to the specifications contained in this publication may be made. Before using this publication in connection with the operation of IBM equipment, contact the local IBM Branch Office for revisions. Because extensive changes have been made throughout this manual, the entire manual should be reviewed.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM General Products Division, Technical Publications, Department G26, 555 Bailey Avenue, San Jose, California 95150. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

©Copyright IBM Corporation 1969, 1983

## Preface

The 2835 Storage Control and 2305 Fixed Head Storage Module form a large capacity, high speed direct access storage facility for general purpose data storage and system residence. It attaches to the central processing unit through a block multiplexer channel, and operates under direct program control of the processor.

For experienced programmers, this manual provides readily-accessible reference material related to channel command words, sense bytes, track format, track capacities, and error recovery.

Less experienced programmers will find sufficient information to create channel programs to best utilize the standard and special features of the facility.

A complete description of the switches and indicators is provided for systems installation operators.

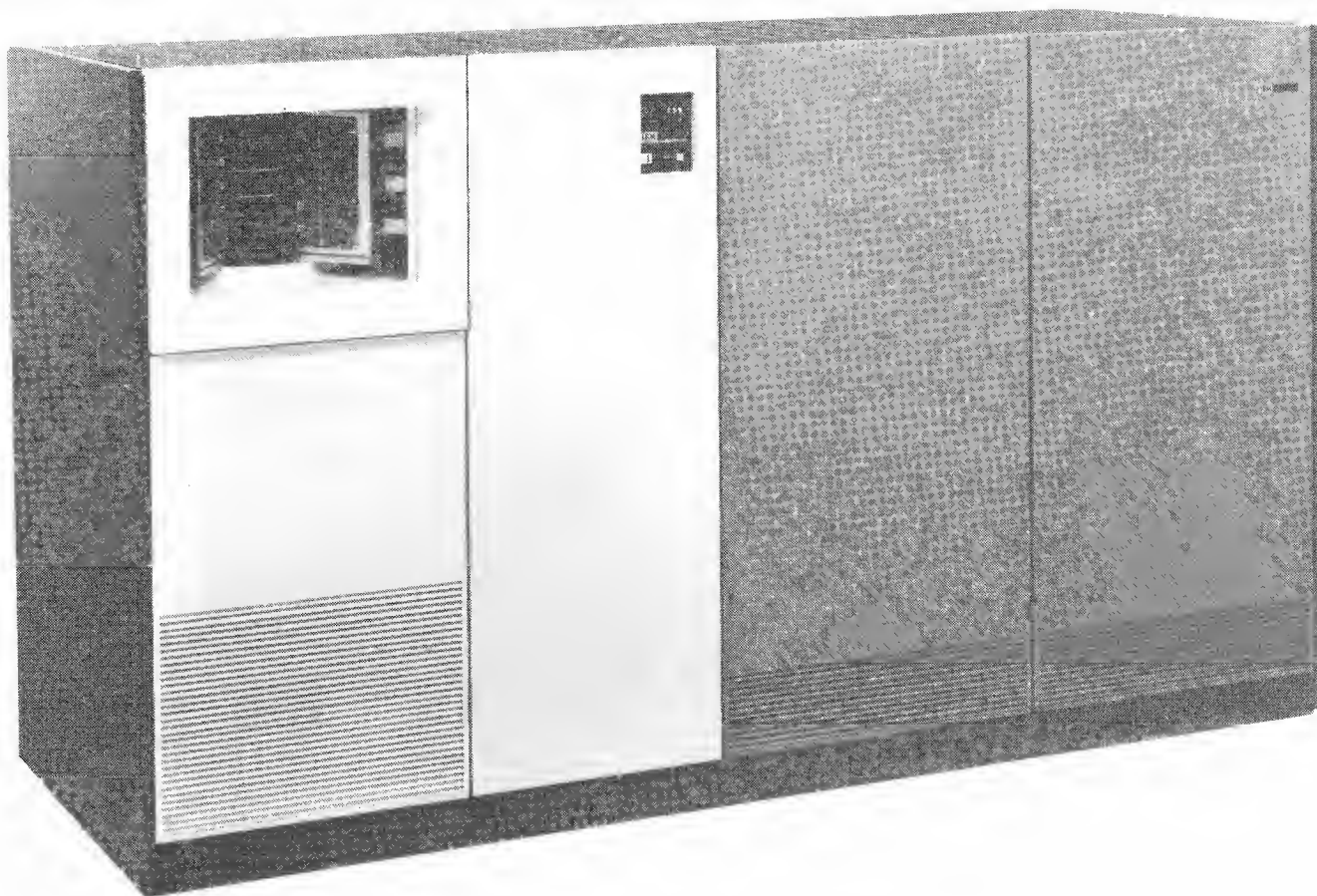
Programmers should be familiar with the information contained in *IBM System/360 Principles of Operation*, Order Number GA22-6821, *IBM System/370 Principles of Operation*, Order Number GA22-7000, *IBM 4300 Processors Principles of Operation for ECPS: VSE Mode*, Order Number GA22-7070, or *IBM 370-XA Principles of Operation*, Order Number SA22-7085, as appropriate. Operators should be familiar with the material presented in the system summary for the parent system. Order numbers for system summary and other related publications can be found in the *IBM System/360 Bibliography*, Order Number GC20-0360, or the *IBM System/370 and 4300 Processors Bibliography*, Order Number GC20-0001.

This major revision of the *IBM 2835/2305 Reference Manual* includes miscellaneous revisions which have accumulated. Information relating to the System/370 Extended Architecture has also been included.

# Contents

<b>2305 Facility</b> . . . . .	1	CCW Formats . . . . .	12
General Characteristics . . . . .	1	Command Codes . . . . .	13
<b>2835 Storage Control</b> . . . . .	1	No-Op . . . . .	14
Capabilities . . . . .	1	Orient . . . . .	15
Data Handling . . . . .	1	Recalibrate . . . . .	16
Control Logic . . . . .	1	Seek . . . . .	17
<b>2305 Fixed Head Storage Modules 1 and 2</b> . . . . .	1	Seek Cylinder . . . . .	18
<b>2305 Model 1</b> . . . . .	1	Seek Head . . . . .	19
General Description . . . . .	1	Space Count . . . . .	20
Speed and Capacity . . . . .	3	Set File Mask . . . . .	21
Operational Description . . . . .	3	Set Sector . . . . .	22
<b>2305 Model 2</b> . . . . .	3	Restore . . . . .	23
General Description . . . . .	3	Vary Sensing . . . . .	24
Speed and Capacity . . . . .	3	Diagnostic Load . . . . .	25
Operational Description . . . . .	3	Diagnostic Write . . . . .	26
<b>Data Characteristics</b> . . . . .	4	Search Home Address Equal . . . . .	27
Record Format . . . . .	4	Search ID Equal . . . . .	28
Address Marker Area . . . . .	4	Search ID High . . . . .	29
Count Area . . . . .	4	Search ID Equal or High . . . . .	30
Key Area . . . . .	4	Search Key Equal . . . . .	31
Data Area . . . . .	4	Search Key High . . . . .	32
Track Format . . . . .	4	Search Key Equal or High . . . . .	33
Record Zero . . . . .	5	Read Home Address . . . . .	34
Data Records . . . . .	5	Read Count . . . . .	35
Gaps . . . . .	5	Read Record Zero . . . . .	36
<b>Error Detection and Correction</b> . . . . .	6	Read Data . . . . .	37
Parity . . . . .	6	Read Key and Data . . . . .	38
Error Correction Code . . . . .	6	Read Count, Key, and Data . . . . .	39
Data Integrity . . . . .	6	Read Initial Program Load . . . . .	40
<b>I/O Operations</b> . . . . .	7	Read Sector . . . . .	41
General Description . . . . .	7	Sense . . . . .	42
Channel Operation . . . . .	7	Read Buffered Log . . . . .	43
Channel Command Word . . . . .	7	Device Release . . . . .	44
Status Presentation . . . . .	7	Device Reserve . . . . .	45
Command Chaining . . . . .	8	Read Diagnostic Status 1 . . . . .	46
Data Chaining . . . . .	8	Write Home Address . . . . .	47
Branching in Channel Programs . . . . .	9	Write Record Zero . . . . .	48
Unit Selection and Device Addressing . . . . .	9	Erase . . . . .	49
<b>Channel Commands</b> . . . . .	10	Write Count, Key, and Data . . . . .	50
Control Commands . . . . .	10	Write Special Count, Key, and Data . . . . .	51
Search Commands . . . . .	10	Write Data . . . . .	52
Read Commands . . . . .	10	Write Key and Data . . . . .	53
Write Commands . . . . .	10	<b>Channel Programs</b> . . . . .	54
Formatting Write Commands . . . . .	10	<b>Standard Features</b> . . . . .	61
Update Write Commands . . . . .	10	Search Operations . . . . .	61
Test I/O . . . . .	11	Multiple Track (MT) Operation . . . . .	61
TIC Command . . . . .	11	Record Overflow . . . . .	61
		Formatting Overflow Records . . . . .	61
		Processing Overflow Records . . . . .	62

End-of-File . . . . .	63	<b>Error Correction Function and Related Tables . . . . .</b>	<b>73</b>
Command Retry . . . . .	64	Error Condition Table . . . . .	73
Multiple Requesting . . . . .	64	Error Correction Function . . . . .	73
Rotational Position Sensing . . . . .	65	Special Cases . . . . .	73
<b>File Protection . . . . .</b>	<b>67</b>	Error Correction Example for Model 2 . . . . .	73
<b>Storage Control Diagnostics . . . . .</b>	<b>68</b>	Construction of Restart CCW . . . . .	73
Usage/Overflow/Error Counters . . . . .	68	Recovery Action Table . . . . .	74
Buffered Log . . . . .	68	<b>Appendix A: Record and Track Capacities . . . . .</b>	<b>76</b>
Disconnect In . . . . .	68	<b>Appendix B: Sense Bytes . . . . .</b>	<b>80</b>
Special Microcodes . . . . .	68	<b>Appendix C: Log Entry Format . . . . .</b>	<b>84</b>
<b>Operator Controls and Indicators . . . . .</b>	<b>69</b>	<b>Index . . . . .</b>	<b>85</b>
<b>Special Features . . . . .</b>	<b>71</b>		
Two Channel Switch . . . . .	71		
Device Reservation . . . . .	71		
Channel Selection Switch . . . . .	71		
Device Status . . . . .	72		
Addressing . . . . .	72		
Resets . . . . .	72		



IBM 2835 Storage Control and 2305 Fixed Head Storage Module

## 2305 Facility

### General Characteristics of the Facility

The IBM 2835 Storage Control and the IBM 2305 Fixed Head Storage Module (briefly and collectively known as the "2305 facility") provide for attachment of fast access, fixed head storage units to the IBM System/360 Model 85 and Model 195, to the IBM System/370 Models 145 through 195, to the IBM 3031, 3032, 3033, 3081, 3083, 3084 Processor Complexes, and to the IBM 4341 Processor. Attachment to System/360 Models 85 and 195, and to System/370 Models 165, 165-II, 168, 168-3, and 195 requires the IBM 2880 Block Multiplexer Channel; otherwise a system block multiplexer channel is required.

Two types of the 2305 facility are available. Both consist of a single 2835 Storage Control and one or two 2305 Fixed Head Storage Modules. In the one type (Model 1) each 2305 utilizes two parallel recording paths, with two recording elements per path. In the other type, (Model 2) each 2305 is equipped with a single recording path and a single recording element per path.

When one or more 2305's are attached to a 2835, the model numbers of all units must match. For example, a 2835 model 1 may control two 2305 model 1's, and 2835 model 2 may control two 2305 model 2's. The 2835 Model 1 can only be attached through a 2880 block multiplex channel with the dual byte interface feature

### 2835 Storage Control

#### Capabilities

The IBM 2835 Storage Control provides the following capabilities:

1. Interpret and execute file commands.
2. Translate and check the integrity of all data moving between the channel byte interface and the module bit interface.
3. Furnish facility status information to the using system.
4. Perform diagnostic evaluation of the facility.

#### Data Handling

**Data Lines:** In the 2835 model 1, two parallel data lines carry information between the 2835 and the 2305. In the 2835 model 2, a single line is used for this purpose. In both models, reading and writing are performed as described in subsequent paragraphs.

**Writing:** The byte(s) received from the channel is serialized and transferred bit-by-bit to the module over the data line(s). Parity is not recorded on the disk. A sync bit, generated by

the control unit, is added to each byte for clocking purposes later when the data is read.

**Reading:** The sync bit is stripped off, and the data bits are assembled into bytes. Parity is generated and added to the data before transmission to the channel.

### Control Logic

The control unit contains a miniature direct access device which provides read-only storage for control logic backup and storage of nonresident microdiagnostics. The recording medium is an inexpensive Mylar\*-coated disk cartridge. This cartridge can be mailed to the customer engineer when changes are required in the functional or maintenance control logic.

### 2305 Fixed Head Storage Module, Models 1 and 2

The IBM 2305 Fixed Head Storage Module is a prime system residence device on large systems that require low access time and high data rates. It meets the requirements of large systems for high speed, high availability auxiliary storage. The 2305 provides capabilities qualitatively similar to those of the IBM 2301 Drum Storage, but both models of the 2305 offer faster access and larger capacity than does the 2301.

Unique features of the 2305 include error correction, vary sensing, alternate track sparing, rotational position sensing, command retry, and multiple requesting, all of which are described in this publication. The sum of these characteristics is a device well adapted for high performance direct access storage applications such as systems residence, paging, index residence, high priority data residence, and high speed work storage.

The functional description of each unit in a basic system configuration is shown in Figure 1.

### 2305 Fixed Head Storage Module, Model 1

#### General Description

The IBM 2305 Fixed Head Storage Module, Model 1, is a high speed auxiliary storage device that provides fast access time and a high speed data rate.

The storage module is a fixed head disk drive module that uses multiple element recording heads. Six disks, permanently mounted in each disk storage module, provide 432 recording tracks; 384 of these tracks are addressable and 48 are spares. Two recording elements are located over each of

\* Trademark of E. I. duPont de Nemours & Co. (Inc.)



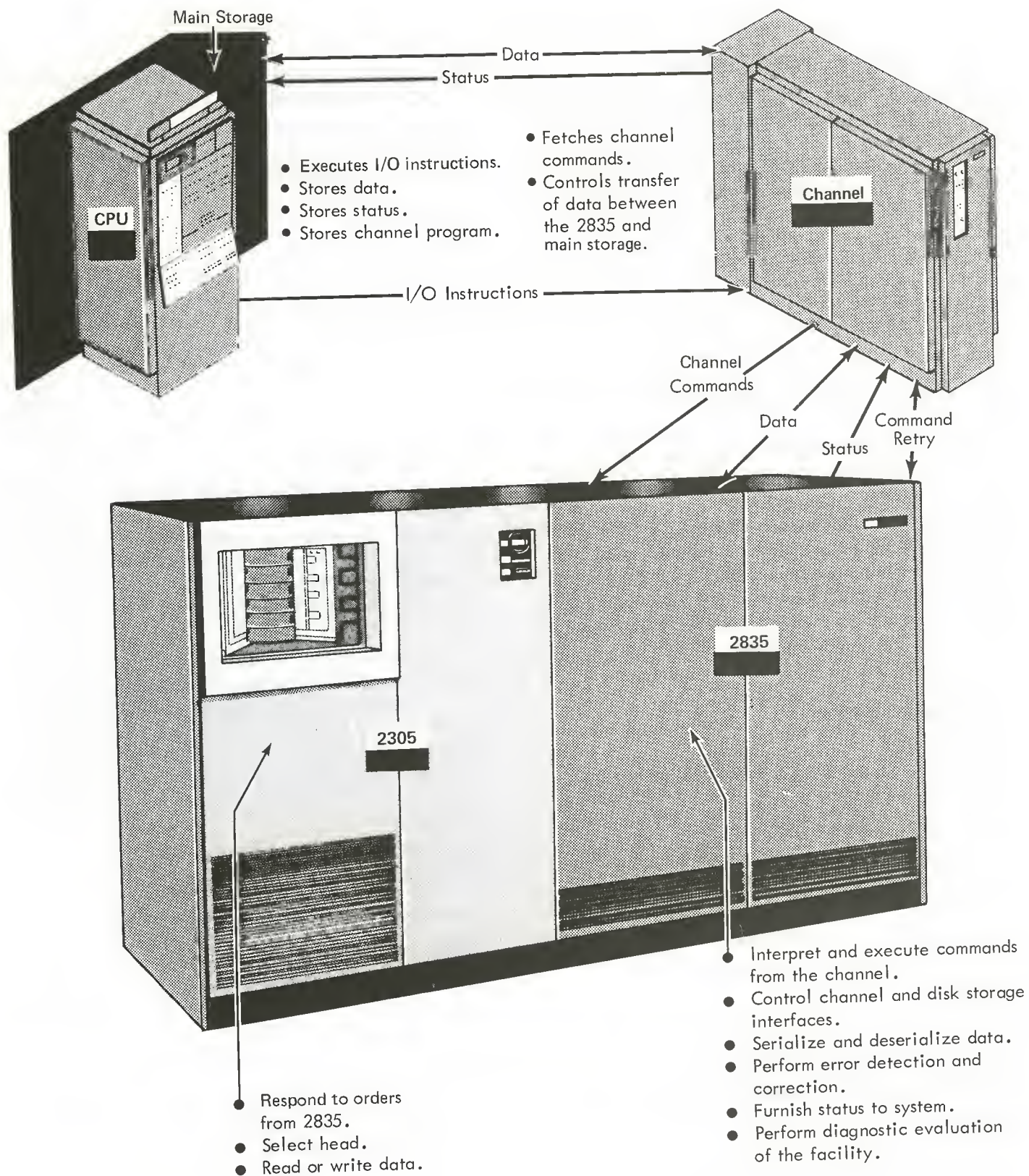


Figure 1. Functional Description

the 384 recording tracks. Average access time is 2.5-millisecond; disk rotational speed is 6000 rpm.

Data is transferred serially by bit (on parallel paths) to the control unit at the rate of 3.0 million bytes/second.

### ***Speed and Capacity***

The following table summarizes 2305 model 1 speed and capacity:

Number of tracks	384
Bytes per track (R0 no key)	14,576
Bytes per track (R1 no key)	14,136
Bytes per module (R1 no key)	5,428,224
Rotation time	10 ms
Access time (maximum)	5.1 ms
Access time (average)	2.5 ms
Transfer rate	3.0 megabytes/sec

### ***Operational Description***

There are two parallel data lines between the control unit and the model 1 disk module. During write operations, the two bytes received in parallel from the channel are serialized and transferred to the disk module. Parity is not recorded; however, Error Correction Code (ECC) bytes are recorded for error detection and correction. During reading, the two channels of serial data are deskewed and assembled into bytes. Parity bits are generated and added before transmission to the channel.

Two recording heads, separated by 180°, are located over each of 384 recording tracks. A recording track contains two logical track segments, each occupying a 180° arc of the recording track. Two of the logical track segments, from opposing disk surfaces, are paired and operated in parallel with their recording heads to form 384 system-addressable tracks. When a track is addressed, the disk module dynamically selects the pair of recording heads that is currently nearest the desired addressed track.

There are 48 spare tracks that can be wired in by the customer engineer to replace defective addressable tracks. One

pair of spare tracks is available for alternate track sparing under system control. To activate one or more of the 48 spare track addresses, an IBM customer engineer manually wires the necessary logic into circuits provided for this purpose. If the 2835 attempts to select a read/write head that has been jumpered out in this manner, the normal address sent by the 2835 is automatically switched to the selected spare head.

## **2305 Fixed Head Storage Module, Model 2**

### ***General Description***

The IBM 2305 Fixed Head Storage Module, Model 2, has the same physical characteristics as the model 1, except that the model 2 has a single head per track. Therefore, there are 768 recording tracks, with an average access time of 5.0 milliseconds.

### ***Speed and Capacity***

The following table summarizes the speed and capacity provided by the model 2 disk module:

Number of addressable tracks	768
Bytes per track (R0 no key)	14,866
Bytes per track (R1 no key)	14,660
Bytes per module (R1 no key)	11,258,880
Rotation time	10 ms
Access time (maximum)	10.25 ms
Access time (average)	5.0 ms
Transfer rate	1.5 megabytes/sec

### ***Operational Description***

The operation is the same as on the model 1, except that bytes are transferred to the channel serially rather than by two bytes in parallel.

There are 864 recording tracks in each module, of which 768 are operational and 96 are spares. Each address is processed through a single data channel.

## Data Characteristics

The basic unit of information recorded on the 2305 facility is eight bits long and is called a byte. A group of bytes separated by a special gap recognized by the 2835 is called an area. Areas of related data are combined to make a record, the logical unit of information.

### Record Format

A record consists of the following four areas: address marker (AM) area, count area, key area, and data area.

#### Address Marker Area

This area is written and used by the 2835 to denote the beginning of a record. The address marker is followed by a gap (G4).

#### Count Area

The 2305 uses self-formatting records; i.e., records which contain information defining their length and format. The count area containing this information is written at the time the record is originally written; this area is not changed until the entire record is rewritten. The process of writing an entire record, including the count area, is called formatting. The size of a record is determined at the time a record is formatted. The count area consists of the following sub-areas and is followed by a gap (G2).

Flag (F)	2 bytes (model 1) or 1 byte (model 2)
Identifier (ID)	5 bytes
Key Length (KL)	1 byte
Data Length (DL)	2 bytes
Error Correction Code Bytes (ECC)	

**Flag:** The first byte of the count area contains the flag. The flag is propagated from record to record by the 2835. The function or setting of each bit in the first flag byte is given in the following list.

- 0 — Unused, written as 0.
- 1 — On if the logical record continues on the next track.  
On for each segment of an overflow record except the last.
- 2-5 — Unused, written as 0.
- 6 — Used for alternate track sparing; bit is not written on track.
- 7 — Unused, written as 0.

The second flag byte is present only on the model 1, and is unused; it is written only because of the parallel recording technique.

**Identifier:** This subarea is used to uniquely identify a record regardless of the content of that record. Because the identifier is not necessarily related to the physical location of the record, it is not considered an address. However, the track address is usually included as a part of the identifier (ID) area. Each ID on a track is usually unique to ensure that each record can be processed without knowledge of its content. The first four bytes of the ID usually consist of the track address; the fifth byte is unique for each record on the track.

**Key Length:** This byte defines the length of the key area. This area is always present; however, it may contain 0, in which case the key area and its gap are omitted.

**Data Length:** These two bytes define the length of the data area. It is always present. If set to 0, it represents the end of the logical file.

**Error Correction Code Bytes:** These bytes are used for error detection and correction in reading the count area.

#### Key Area

The length of this area is defined by the key length (KL) subarea. Because correction code bytes are added to this area, it is actually longer than the length given in the key length. Once the area is formatted, its contents, (but not its length) can be altered.

The key area is used to identify the information recorded in the data area of a record. If the key area is altered, the data area of the record must be rewritten as well. The key area is followed by a gap (G2).

#### Data Area

The length of this area is defined by the data length (DL) subarea. Because correction code bytes are added to this area, it is actually longer than the length given in the data length. Once the area is formatted, its contents, but not its length, can be altered. The data can be altered without affecting any other area in the record. The data area is followed by a gap (G3 or G5).

### Track Format

A track (Figure 2) is the smallest physically addressable recording area on the 2305. Each addressable track has a unique physical address. One or more records are written on each track. The first record following the index point is always record zero (R0).

The 2305 does not record a home address area between the index point and R0. However, for compatibility with other similar storage devices, it does accept and emulate Write

Home Address, Read Home Address, and Search Home Address commands.

### Record Zero

Record zero may be used as a normal data record. However, it is usually reserved by the operating system for nonuser functions. R0 differs from all other records in the following ways:

1. Every addressable track contains only one R0.
2. R0 is always the first record on a track following index.
3. Special commands are provided which operate only on R0.
4. R0 is not preceded by an address marker.

### Data Records (R1-Rn)

One or more data records may follow R0 on a track. Count areas make each record self-formatting for maximum data organization, flexibility, and efficiency.

### Gaps

The record gap is a string of bytes written by the 2835 to delimit records and areas within records. Gaps are not accessible to the using system nor under its control. There are five gap types, designated G1 through G5. G5 is variable in length; the other four are fixed in length.

**Gap 1:** This gap is written between index and the R0 count area.

**Gap 2:** This gap is written between the count area and the key area, and between the key area and the data area.

**Gap 3:** This gap is written between the data area of one record and the address marker of the following record.

**Gap 4:** This gap is written between the address marker and the count area of every record except R0.

**Gap 5:** This gap is written following the data area of the last record on the track. It is variable in length and extends to the index point.

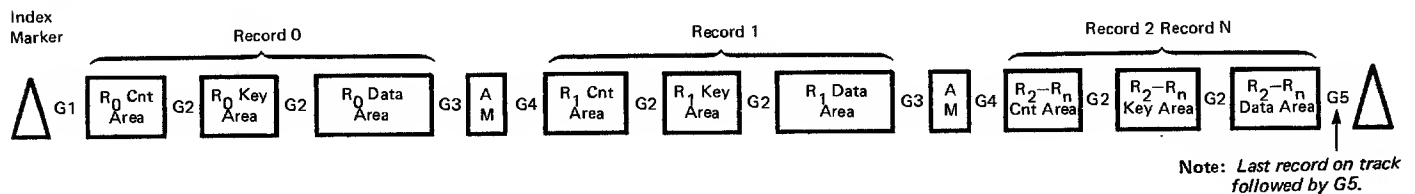


Figure 2. Track Format

## Error Detection and Correction

### Parity

To check data accuracy, a parity bit is associated with each byte within the processor and channel. When a byte is formed, the parity bit is set to either 1 or 0 to maintain an odd number of 1 bits within the byte (odd parity). Each byte of data to be written is checked for correct parity as it is received by the 2835.

### Error Correction Code

As data is being transferred from the channel to the disk storage (written), the storage control removes the parity bit associated with each byte. It then computes error correction code bytes which are written after each recorded area. The correction code bytes are coded to represent the data in the recorded area, and are used for both error detection and correction.

As the data is being transferred from the disk storage to the channel (read), each area is inspected by the storage control and the error correction code bytes are recalculated for each area. The 2835 correction code can detect single burst errors of 470 bits or less, and correct single bursts of 11 bits or less.

The correction code bytes are removed and proper parity is generated by the storage control before the data is transferred to the channel.

If a correctable data error is detected in the count or key areas, the storage control internally executes the error correction function through use of command retry. (See "Command Retry.") If a correctable data error in a data area is detected, the operation is terminated and the correction function is performed by the system error recovery procedures. (See "Error Recovery Procedures.")

### Data Integrity

Unless corrected immediately, soft write errors may cause hard read errors. Therefore, where data integrity is required, verification should be incorporated within the program. Then, in the event of soft errors, the record can be rewritten and verified before the original data has been destroyed.

Two methods can be used for verification; the full read-back check and the correction code check.

**Full Read-Back Check:** All of the data just written is read back into main storage and compared with the original information byte for byte.

**Correction Code Check:** A read operation is performed with the skip bit on. In this method the control unit checks the validity of the record by using the error correction code bytes.



# Input/Output Operations

## General Description

I/O operations, initiated by I/O instructions in the system control program, are controlled by commands fetched from main storage by the channel. Arithmetical and logical decision operations are performed while the processing unit is in either the supervisor state or the problem state; I/O operations require the processing unit to be in the supervisor state.

The processing unit is changed from problem to supervisor state when a supervisor call instruction is executed or an I/O interrupt occurs. The status of the system existing at the time of the change is stored in the program status word.

After an I/O instruction has been executed, the processor can return to the problem state and continue the interrupted program by reloading the program status word stored when the program entered the supervisor state.

## Channel Operation

After successful execution of an I/O instruction, the channel selects and governs the addressed storage control and drive. Main storage locations contain information that enables the channel to perform the functions necessary to complete the operation.

### *Channel Command Word*

The channel fetches the first channel command word (CCW) from the address specified by the control program. The channel command word specifies the operation to be performed, the main storage location to be used, and the action to be taken when the operation is completed.

If the channel is available when it receives the channel command word, it attempts to select the device specified in the I/O instruction by sending the address to all attached control units. If the addressed device is attached to the channel and has power on, the command code portion of the channel command word is sent to the storage control. The storage control responds with an initial status byte to the channel.

The results of the attempt to initiate execution of the command may be indicated by a condition code. If the I/O operation was not started, new status information to show why is normally set in the channel.

### *Status Presentation*

Status is presented twice (initial status and ending status) for all commands except immediate commands not chained from write commands.

Immediate commands present channel end and device end in initial status. An immediate command chained from a write command causes channel end status to be presented in initial status and device end is presented in ending status when the write operation is finished.

Figure 3 shows the format and content of the device status byte.

**Initial Status:** The initial status byte is zero for the test I/O command and all non-immediate commands unless one or more of the following conditions exists.

1. The storage control is busy.
2. A status condition is pending. See "Pending Status."
3. A unit check condition exists.
4. Command retry is required. See "Command Retry."

**Ending Status:** In most cases, channel end and device end are presented as the normal ending sequence for an operation. However, on block multiplexer channels, channel end alone may be treated by the channel as ending status with the device end treated as pending status, thus allowing the channel to disconnect and initiate activity to other devices.

If an error has occurred during the operation, unit check will accompany the channel end-device end status.

**Pending Status:** A pending status condition may exist for either the storage control or a device.

Status is pending for the storage control if:

1. An interface disconnect sequence was signaled after a command was issued, but before channel end status was accepted.
2. Busy, channel end, or unit check status was stacked by the channel.
3. Zero status in response to a test I/O command was stacked by the channel.
4. Control unit busy was presented to the channel. (Control unit end is pending.)
5. Unit check was detected for an operation after device end had been cleared.

Status pending for the storage control (except for control unit end) causes the storage control to appear busy for all exposures\* except the exposure for which the status condition exists. Unless it is busy, the storage control will request

\* The term "exposure" is used in this publication to identify a logical device address. Logical addresses are 0-7 for the first 2305 module and 8-15 for the second 2305 module.

CSW Bit Position	Field Designation	Function
0	Attention	Not used by 2835.
1	Status Modifier	<p>Set whenever a Search High, Search Equal, or a Search High or Equal command has been executed and the condition satisfied.</p> <p>The status modifier is also set whenever the storage control is busy. This bit, in conjunction with the busy bit, signifies control unit busy.</p> <p>Status modifier set with unit check and channel end (or channel end and device end) indicates that a retrievable error has been encountered. The storage control automatically retries the command.</p>
2	Control Unit End	Set if a control unit busy status has been generated previously and the busy condition has been terminated.
3	Busy	<p>Indicates that the selected device is busy.</p> <p>In conjunction with the status modifier bit indicates the storage control is busy. It is set when a new command chain cannot be initiated because the storage control is causing a track to be erased following a Format Write command.</p>
4	Channel End	Set at the end of each channel command.
5	Device End	Indicates that the channel is available to interrupt.
6	Unit Check	<p>Set whenever an unusual or error condition is detected.</p> <p>A Sense command may then be used to identify the condition.</p>
7	Unit Exception	<p>Indicates an end-of-file has been detected during a read RO, read IPL, read CKD, read KD, read D, write KD, or a write D operation. It results from a data length of zero being detected in the count area of a record. When this condition is detected no data is transferred from the data area. If the key length is not zero, the key area is transferred.</p>

Figure 3. Device Status

service to clear the pending status condition. Status is cleared when presented to, and accepted by, the channel.

Status is pending for a device if:

1. Channel end was presented alone.
2. Busy status was presented.
3. The drive has gone from not ready to ready.

Status pending for a device causes the storage control to request service when both the storage control and device are not busy. The status is cleared when presented to, and accepted by, the channel.

**Contingent Connection:** A contingent connection is established in the storage control after the channel accepts a status byte containing unit check. It lasts until a command other than test I/O or no-op receives an initial status byte of zero for the storage control and exposure that generated the unit check or a selective or system reset occurs.

During the contingent connection state, the storage control appears busy to all device addresses other than the address for which the contingent connection was established.

### Command Chaining

It is typical for the 2835/2305 to execute a series of channel commands in a single channel program. This method of operation is called command chaining. Command chaining is initiated by turning on a flag bit in the channel command word. The channel fetches a new CCW, specifying a new I/O operation, on completion of the current CCW. The new I/O operation is automatically executed when the 2305 has completed the current operation and signaled device end to the channel.

The completion of the current CCW does not cause an I/O interruption condition, and the count indicating the amount of data transferred is not available to the program.

Command chaining is normally used with all 2835/2305 channel programs. Time is made available to execute command chaining functions in the gap area between record areas.

Certain restrictions exist regarding sequences of commands within chains. These restrictions are discussed with the individual command descriptions in the "Channel Commands" section of this manual.

### Data Chaining

Data transferred between main storage and the 2305 may be chained. Data chaining permits blocks of data to be transferred to or from noncontiguous areas of main storage. When data chaining is specified by a flag in the CCW, the channel fetches a new CCW, specifying a new storage location, upon completion of data transfer for the current channel

command word. Unless the command code specifies transfer in channel, the command code of the new CCW is ignored.

Data chaining may be used to rearrange information as it is transferred between main storage and the 2305. It may also be used in conjunction with the skip flag to enable the program to place selected portions of a block of data in main storage.

Data chaining occurs immediately after the last byte of data designated by the current CCW has been transferred to main storage or been accepted by the 2305.

If both data chaining and command chaining are indicated in the channel command word, data chaining takes precedence and command chaining is ignored.

**Note:** *Because of the high data rate of the 2305, read or write data chaining within record areas may cause unpredictable overruns or chaining checks. The following conditions are particularly sensitive.*

- Simultaneous use of data chaining and the PCI flag.
- Data chaining with small byte counts and the CCW data address not on a word boundary.
- On the model 1, data chaining with an odd byte count that is not equal to the length of the record area.

#### ***Branching in Channel Programs***

Normally, the next CCW in a chain of channel commands is taken from an address eight positions higher than the address of the current CCW. This sequence can be modified in either of two ways:

1. If command chaining is specified in a search command, and execution of the command results in a status modifier indication (search satisfied), the channel fetches the next CCW from a main storage location sixteen positions higher than the current channel command.
2. The transfer in channel command (TIC) may be used to modify the sequence of a chain of commands. The data address portion of the TIC CCW specifies the main storage location of the next channel command word. Therefore, the next CCW may be fetched from any valid main storage location.

These methods of modifying the sequence of a CCW chain provide branching capabilities within a channel program.

#### ***Unit Selection and Device Addressing***

The device address consists of an eight-bit byte. This information is used to specify the storage control and the device.

The four high-order bits specify the storage control. Because the storage control address is plugged into an address card at installation time, these four bits may be any configuration.

The four low-order bits specify the device. A maximum of two modules can be attached to the storage control. Therefore, the addresses are restricted to 0 and 8. However, to support multiple requesting, the device address bits may have any value from 0 to 15. A device address bit in the 0-7 group causes the control unit to select the module with physical address 0. An address in the group 8-15 selects the module with physical address 8.



## Channel Commands

### Control Commands

Control commands do not involve a transfer of data records between the storage control and main storage. However, in certain operations, control bytes are transferred from main storage to the storage control to enable the operation to take place. The control bytes are parity checked during transfer.

### Search Commands

On all search commands, the channel operates in the write mode while the disk storage operates in the read mode. The storage control compares the data from main storage against the data from the drive. When the search criteria has been satisfied (compared equal, high, etc.) the storage control returns a status modifier bit with channel end and device end. The status modifier bit causes the channel to skip the next CCW in the chain and fetch the next command from a storage location 16 positions higher than the current CCW.

Each search command operates only on one record. To search another record, the search command must be re-issued. This is normally done by chaining a transfer in channel command to the search command. The following is an example of this procedure:

Search Key Equal

TIC \* -8

Read Data

As long as the search is unsuccessful, the transfer in channel command following the search command causes the search to be repeated. When the search is successful, the status modifier causes the TIC command to be skipped and the read data command is executed.

At the end of every field searched, the validity of the data read is verified by the correction code bytes following the search field. After the correction code check, appropriate ending status is generated and presented to the channel.

If a data overrun or data check is detected, the storage control attempts recovery through use of command retry. If command retry is unsuccessful, channel end, device end, and unit check status is presented.

### Read Commands

A read command is used to transfer information from the disk storage to main storage. Read commands may operate in either single track or multiple track mode.

**Note:** *Read IPL and read sector do not operate in multi-track mode.*

For all read commands, the storage control checks the validity of each area of each record as it is transferred from the disk storage to the storage control. After the correction code bytes have been examined and the validity of the data is established, the storage control sends an ending status byte of channel end and device end to the channel.

If a data overrun or data check is detected, the storage control attempts recovery through use of command retry. If command retry is unsuccessful, or not used, channel end, device end, and unit check are presented to the channel.

### Write Commands

#### *Formatting Write Commands*

Formatting write commands are used to initialize tracks and records, and establish the length of the areas within each record. Error correction code bytes are calculated and written after each area of a record.

Formatting write commands are:

- Write home address.
- Write R0.
- Write count, key, and data.
- Write special count, key, and data.
- Erase.

The command prerequisites and file mask settings for these commands are very explicit and any violation prevents command execution.

Format write commands may be chained together if each satisfies the required prerequisites. After the last format write command in a chain has been completed, the storage control causes the remaining portion of the track to be erased.

If a command other than a format write command is chained from a format write command, it is executed after the track has been erased. If a new command chain is attempted before the end of the track is reached, a short control unit busy sequence (busy and status modifier bits) is presented to the channel. In this case, a control unit end signal is generated at the end of the track.

#### *Update Write Commands*

Update (non-formatting) write commands are used to update existing records, and must operate on previously formatted tracks. Error correction code bytes are calculated and written after each key and/or data area in the record. Update write commands are:

- Write Data.
- Write Key and Data.

If a data overrun occurs during an update write operation, the storage control attempts recovery through use of command retry. If the retry is unsuccessful, channel end, device end, and unit check status is presented to the channel.

### Test I/O

This command is used to determine the status of the 2305 facility and identify any specific errors or unusual conditions that have occurred.

*Note: Since the Test I/O command is not the result of the channel executing a CCW, its operation is explained at this time instead of with the other channel commands. A Test I/O command (00) is not written by the programmer. A command code of all zeros specified in a CCW is considered invalid and causes a program check.*

The Test I/O command is generated automatically by the channel when the channel requires status information. It appears to the storage control as a command byte of all zeros and is treated as an immediate command. Test I/O requests the storage control to send outstanding status information to the channel and normally presents an all zero status byte. Stacked or pending status (if any) is presented in initial status.

### TIC Command

The assembler language notation TIC \*-8 indicates an unconditional branch to the TIC storage address (\*) minus a count of eight. TIC \*-16 indicates an unconditional branch to the TIC storage address (\*) minus a count of sixteen.

The channel command word consists of eight bytes. Two formats are used, depending upon the processor and the program application. The two formats are shown in Figure 4. Individual commands are described in the following pages of this section.

0 7 8		31 32		39 40		47 48		63	
Command Code	Data Address	Flags	Not Used	Count					

0		7 8		15 16		31 32		63	
Command Code	Flags		Count		Data Address				

### Figure 4. Channel Command Word Formats

# Command Codes

Command	Command Code				
	Multiple Track OFF		Multiple Track ON (if applicable)		Page
	Hexadecimal	Binary	Hexadecimal	Binary	
No Operation	03	0000 0011			14
Orient	2B	0010 1011			15
Recalibrate	13	0001 0011			16
Seek	07	0000 0111			17
Seek Cylinder	0B	0000 1011			18
Seek Head	1B	0001 1011			19
Space Count	0F	0000 1111			20
Set File Mask	1F	0001 1111			21
Set Sector	23	0010 0011			22
Restore	17	0001 0111			23
Vary Sensing	27	0010 0111			24
Diagnostic Load	53	0101 0011			25
Diagnostic Write	73	0111 0011			26
Home Address Equal	39	0011 1001	B9	1011 1001	27
Identifier Equal	31	0011 0001	B1	1011 0001	28
Identifier High	51	0101 0001	D1	1101 0001	29
Identifier Equal or High	71	0111 0001	F1	1111 0001	30
Key Equal	29	0010 1001	A9	1010 1001	31
Key High	49	0100 1001	C9	1100 1001	32
Key Equal or High	69	0110 1001	E9	1110 1001	33
Home Address Count	1A	0001 1010	9A	1001 1010	34
Record 0	12	0001 0010	92	1001 0010	35
Data	16	0001 0110	96	1001 0110	36
Key and Data	06	0000 0110	86	1000 0110	37
Count, Key, and Data	0E	0000 1110	8E	1000 1110	38
IPL	1E	0001 1110	9E	1001 1110	39
Sector	02	0000 0010			40
	22	0010 0010			41
Sense	04	0000 0100			42
Read Buffered Log	24	0010 0100			43
Release *	94	1001 0100			44
Reserve *	B4	1011 0100			45
Read Diagnostic Status 1	44	0100 0100			46
Home Address	19	0001 1001			47
Record 0	15	0001 0101			48
Erase	11	0001 0001			49
Count, Key, and Data	1D	0001 1101			50
Special Count, Key, and Data	01	0000 0001			51
Data	05	0000 0101			52
Key and Data	0D	0000 1101			53

\*Two-channel switch feature.

C  
O  
N  
T  
R  
O  
L

S  
E  
A  
R  
C  
H

R  
E  
A  
D

S  
E  
N  
S  
E

W  
R  
I  
T  
E

## NO-OP

Command Code	Data Address	Flags	Count
Binary <b>0000 0011</b> Hex <b>03</b>	Not checked for validity; should not exceed addressing capacity.	Used at the discretion of the programmer.	Must be non zero for format 0 to avoid program check.

**NO-OP** an immediate command, causes no action at addressed device.

**CHANNEL END** is presented in initial status.

**DEVICE END** normally presented in initial status, is presented in ending status if device is completing a format write operation.

**INDISCRIMINATE USAGE** must be avoided; a No-op resets orientation information causing all or part of the records to be skipped.

**EXAMPLE:** a No-op inserted between Read Count and Read Data causes the following record's data to be read.

**EXAMPLE:** a No-op inserted between a command which reads the data field of record n-1 and a command which must process the count area of record n, may skip record n and process the count area of record n+1.

**NO-OP CCW** count field must not be zero for format 0.

## ORIENT

Command Code	Data Address	Flags	Count
Binary <b>0010 1011</b> Hex <b>2B</b>	Allows control of head selection during command execution.	Used at the discretion of the programmer.	Must be non-zero for format 0 to avoid program check.

### MODEL 1

**ORIENT** a data recovery tool; does not transfer data to or from the channel. This command allows control of head selection during command execution.

**INITIAL STATUS** is zero.

**COMMAND EXECUTION** initially orients the track at index with head A selected; head B is selected at next half rotation.

**READ R0 CCW** chained to Orient causes R0 to be read using head A.

**CHANNEL END/DEVICE END** presented to channel after index is detected.

### MODEL 2

**COMMAND EXECUTION** causes no action.

**CHANNEL END/DEVICE END** presented in initial status.

*Note: If command is chained from a format write command:*

**CHANNEL END** is presented in initial status.

**DEVICE END** presented when index is detected.

## RECALIBRATE

Command Code	Data Address	Flags	Count
Binary <b>0001 0011</b> Hex <b>13</b>	Not checked for validity, but should not exceed addressing capacity.	Used at the discretion of the programmer.	Must be non-zero for format 0 to avoid a program check.

**RECALIBRATE** causes no action — maintained for compatibility with other direct access storage devices.

**CHANNEL END/DEVICE END** presented in initial status.

**CCW COUNT FIELD** must not be zero for format 0.

## SEEK

Command Code	Data Address	Flags	Count
Binary <b>0000 0111</b> Hex <b>07</b>	Specifies main storage location of the seek address.	Used at the discretion of the programmer.	Six

**SEEK** transfers the six-byte seek address from channel to storage control.

**INITIAL STATUS** normally zero.

**STORAGE CONTROL** selects drive and proper head.

**CCW COUNT  $\geq$  SIX** transfers six bytes of address information.

**CCW COUNT  $<$  SIX** Seek command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**VALID SEEK ADDRESS** checked by storage control.

**BYTES 0, 1, 2, and 4** must be zero.

**BYTE \*3** must not exceed 48 (decimal) on a model 1 or 96 (decimal) on a model 2.

**BYTE 5** must not exceed 7.

**INVALID SEEK ADDRESS** Seek command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**PARITY ERROR** detected in transfer of seek address: command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates bus out parity error.

**COMMAND EXECUTION** does not require preceding CCW.

**FILE MASK** must allow seeks.

**CHANNEL END/DEVICE END** presented after transfer of seek address.

\* Cylinder 48 (decimal), head 0 is the alternate track spare for Model 1.  
Cylinder 96 (decimal), head 0 is the alternate track spare for Model 2.



## SEEK CYL

Command Code	Data Address	Flags	Count
Binary <b>0000 1011</b> Hex <b>0B</b>	Specifies main storage location of the seek address.	Used at the discretion of the programmer.	Six

**SEEK CYLINDER** transfers the six-byte seek address from channel to storage control.

**INITIAL STATUS** normally zero.

**STORAGE CONTROL** selects drive and proper head.

**CCW COUNT  $\geq$  SIX** transfers six bytes of address information.

**CCW COUNT  $<$  SIX** Seek Cylinder command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**VALID SEEK ADDRESS** checked by storage control.

Bytes 0, 1, 2, and 4 must be zero.

\*Byte 3 must not exceed 48 (decimal) on model 1 or 96 (decimal) on model 2.

Byte 5 must not exceed 7 (decimal).

**INVALID SEEK ADDRESS** Seek Cylinder command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**PARITY ERROR** detected in transfer of seek address: command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates bus-out parity error.

**COMMAND EXECUTION** does not require preceding CCW.

**FILE MASK** must allow Seek Cylinder commands.

**CHANNEL END/DEVICE END** presented after transfer of seek address.

\* Cylinder 48 (decimal), head 0 is alternate track spare on model 1.

Cylinder 96 (decimal), head 0 is alternate track spare on model 2.

## SEEK HEAD

Command Code	Data Address	Flags	Count
Binary <b>0001 1011</b> Hex <b>1B</b>	Specifies main storage location of seek address.	Used at the discretion of the programmer.	Six

**SEEK HEAD** transfers seek address from channel to storage control.

**INITIAL STATUS** normally zero.

**CCW COUNT  $\geq$  SIX** transfers six bytes of address information.

**CCW COUNT  $<$  SIX** Seek Head command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**VALID SEEK ADDRESS** (Six byte) required; however, only the head address specified in the sixth byte is significant; i.e. another valid cylinder address is ignored.

Bytes 0, 1, 2, and 4 must be zero.

\*Byte 3 is not checked. Normally it should not exceed 48 (decimal) on model 1 or 96 (decimal) on model 2.

Byte 5 must not exceed 7 (decimal).

**INVALID SEEK ADDRESS** Seek Head command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates command reject.

**PARITY ERROR** detected in transfer of seek address; command is not executed; unit check, channel end, and device end are presented in ending status. A subsequent Sense command indicates bus out parity error.

**COMMAND EXECUTION** does not require preceding CCW.

**FILE MASK** if preceding seek must be set to allow head seeks.

**CHANNEL END/DEVICE END** presented after transfer of seek address.

\* Cylinder 48 (decimal), head 0 is alternate track spare on model 1.  
Cylinder 96 (decimal), head 0 is alternate track spare on model 2.

## SPACE COUNT

Command Code	Data Address	Flags	Count
Binary 0000 1111 Hex 0F	Specifies main storage location of the key and data lengths of record to be recovered.	Used at the discretion of the programmer.	Three

**CHAINING AND SPECIAL REQUIREMENTS**

1. Cannot be chained from a format Write, Erase, or Space Count command.
2. Must not be followed by a Write, Erase, or Set File Mask command in the same chain.

**SPACE COUNT** allows bypassing a defective count area on a track for recovering data in key and/or data areas following the defective area.

**DATA TRANSFERRED FROM CHANNEL** is used by the storage control as the key length (first byte) and data length (last two bytes) of the record to be recovered.

**CCW COUNT  $\geq$  THREE** three bytes are transferred.

**CCW COUNT  $<$  THREE** specified number of bytes is transferred.

**NO BYTES TRANSFERRED** storage control assumes a value of zero.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented to the channel.

*Note: If Space Count is chained to a command that reads the count area, the count field is ignored. Therefore, an incorrect length indication should be suppressed.*

## SET FILE MASK

Command Code	Data Address	Flags	Count
Binary 0001 1111 Hex 1F	Specifies main storage location of mask byte.	Used at the discretion of the programmer.	One

**CHAINING AND SPECIAL REQUIREMENTS** no more than one Set File Mask command permitted in a CCW chain.

**SET FILE MASK** sets the write and seek masks which provide protection for 2305 data and defines command retry-PCI interaction.

Bit 0	Bit 1	Function	Bit 3	Bit 4	Function	Bit 5	Function	Bit 6	Bit 7	Function
0	0	Inhibit Write Home Address and Write R0.	0	0	Permit all Seek commands.	0	Inhibit Diagnostic Write Command.	0	x	Permit command retry.
0	1	Inhibit all write commands.	0	1	Permit Seek Cylinder and Seek Head.	1	Permit Diagnostic Write Command.	1	0	Inhibit command retry.
1	0	Inhibit all format write commands.	1	0	Permit Seek Head.			0	1	Interrupt after successful retry.
1	1	Permit all write commands.	1	1	Inhibit all seek commands and head switching.					

**COMMAND EXECUTION** is allowable only once within a CCW chain. An attempt to issue more than one Set File Mask in a CCW chain causes a unit check in initial status.

**COMMAND REJECT** is indicated by a subsequent Sense command.

**FILE MASK RESET** to 0's at end of CCW chain.

**WRITE COMMANDS** that violate file mask are not executed.

**UNIT CHECK** is presented in initial status.

**COMMAND REJECT** is indicated by a subsequent Sense command.

**SEEK COMMANDS** that violate the file mask are not executed.

**UNIT CHECK** is presented in initial status.

**FILE PROTECTED** is indicated by a subsequent Sense command.

**MULTI-TRACK/OVERFLOW** operations that violate the file mask are indicated with unit check and file protected.

**CHANNEL END/DEVICE END** are presented to the channel at completion of the mask byte transfer.

**SYSTEM OR SELECTIVE RESET** resets the file mask to 0's.

A new command chain after a reset without a set file mask CCW, permits Seek and Write commands (except Write Home Address and Write R0).

## SET SECTOR

Command Code	Data Address	Flags	Count
Binary <b>0010 0011</b> Hex <b>23</b>	Specifies main storage location of desired sector.	Used at the discretion of the programmer.	One

**SET SECTOR** used on block multiplexer channels; eliminates the necessity to maintain channel connection during rotational delay.

In selector channel mode, the channel remains connected during rotational delay.

**COMMAND EXECUTION** transfers a sector number from main storage control.

**SECTOR VALUES** are checked for validity by the 2835.

- \* **VALID ARGUMENT** (Model 1) (0-89): 1. Storage control presents channel end and disconnects.
- \* **VALID ARGUMENT** (Model 2) (0-179): 2. Device end is signaled via a polling sequence when angular position is reached and channel reconnects to continue chain.  
3. If reconnection does not occur, the storage control attempts reconnection on subsequent revolutions.

### **ZERO ARGUMENT:**

Storage control attempts reconnection prior to index.

Channel end, device end, and unit check presented in ending status.

### **ARGUMENT = 255:**

1. Command is treated as a No-op.
2. Channel end/device end presented in ending status.
3. Track orientation is destroyed.

\* All valid arguments are adjusted by the storage control to compensate for channel reselection delay.

## RESTORE

Command Code	Data Address	Flags	Count
Binary <b>0001 0111</b> Hex <b>17</b>	Not checked for validity; must not exceed addressing capacity.	Used at the discretion of the programmer.	Must be non-zero for format 0 to avoid a program check.

**RESTORE** is maintained primarily for compatibility with other IBM Direct Access Storage Devices. No action is performed at the 2305.

**CHANNEL END/DEVICE END** presented in initial status.

## VARY SENSING

Command Code	Data Address	Flags	Count
Binary <b>0010 0111</b> Hex <b>27</b>	Specifies the storage location of a byte which contains control information to vary read sensing level.	Used at the discretion of the programmer	One

**VARY SENSING** attempts data recovery following an uncorrectable data check.

**INITIAL STATUS** is zero.

**COMMAND EXECUTION** causes one byte of data to be transferred from the system to storage control.

**STORAGE CONTROL** sends clip level and clip path to module.

### BYTE FORMAT:

Bit(s)	Function
0-2	Not used
3	Clip plus (clip minus if zero)
4,5,6	Not used
(Model 1) 7	Path 1 (path 0 if zero)
(Model 2) 7	Zero

**CHANNEL END/DEVICE END** presented to channel after byte is transferred.

## DIAGNOSTIC LOAD

Command Code	Data Address	Flags	Count
Binary <b>0101 0011</b> Hex <b>53</b>	Specifies main storage location of control byte.	Used at the discretion of the programmer.	One

**DIAGNOSTIC LOAD** transfers a 512 byte block of data from storage control read-only storage to storage control buffer.

**DATA BLOCK** transferred is a functional control logic diagnostic test.

**CHAINING AND SPECIAL REQUIREMENTS** See Note 1 below.

**INITIAL STATUS** normally zero.

**CONTROL BYTE** specifying diagnostic control logic ID number, is transferred from main storage to storage control.

**TRACK ADDRESS** (0-31) is specified by bits 0-4. See Note 2.

**SECTOR NUMBER** (0-7) is specified by bits 5-7. See Note 2.

**STORAGE CONTROL** transfers diagnostic test to buffer.

**DATA TRANSFER COMPLETE** causes storage control to request service and present channel end and device end when polled.

**COMMAND EXECUTION** allows any drive address to be used with the storage control address.

**READ DIAGNOSTIC STATUS 1** command transfers the diagnostic test from storage control buffer to main storage.

**Note 1:** *This command will not execute with 2835 microcode for basic and two channel switch (P/N 2254580) or four channel switch RPQ (P/N 2254581). These two microcode loads are used only for attachment to the 3081 Model D. Using this command with these microcodes will cause a unit check with sense byte 7=C1 to be returned.*

**Note 2:** *Track address and sector number are references to the read-only storage device located within the 2835, not to a 2305 disk drive.*



## DIAGNOSTIC WRITE

Command Code	Data Address	Flags	Count
Binary <b>0111 0011</b> Hex <b>73</b>	Specifies main storage location of diagnostic test.	Used at the discretion of the programmer.	512

**CHAINING AND SPECIAL REQUIREMENTS** must be preceded in the chain by a Set File Mask command with bit 5 of the mask byte set to 1. See Note below.

**DIAGNOSTIC WRITE** transfers a 512 byte diagnostic test from main storage to storage control.

**INITIAL STATUS** normally zero.

**DATA TRANSFER COMPLETE** test execution begins.

**TEST COMPLETE** 16 byte error code message is stored in storage control buffer.

**CCW COUNT  $\geq$  512** 512 bytes are transferred.

**CCW COUNT  $<$  512** only the specified number of bytes are transferred, command is terminated and channel end, device end, and unit check are presented in ending status. A subsequent Sense command indicates command reject.

**ERROR CODE MESSAGE** (16 bytes) is transferred from storage control buffer to main storage by subsequent Read Diagnostic Status 1 command.

**CHANNEL END & DEVICE END** presented after test complete.

**Caution:** This command is intended for maintenance purposes only. Any use other than that provided by IBM diagnostic programs may yield unpredictable results.

**Note:** This command will not execute with 2835 microcode for basic and two channel switch (P/N 2254580) or four channel switch RPQ (P/N 2254581). These two microcode loads are used only for attachment to the 3081 Model D. Using this command with these microcodes will cause a unit check with sense byte 7=C1 to be returned.

## SEARCH HOME ADDRESS EQUAL

Command Code	Data Address	Flags	Count
Binary <b>0011 1001</b> Hex <b>39</b> Multitrack 1011 1001 Hex B9	Specifies main storage location of a cylinder number (CC) and head number (HH).	Used at the discretion of the programmer.	Four

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**INITIAL STATUS** normally zero.

**CYLINDER/HEAD NUMBERS** from main storage and current head address are compared by storage control immediately. Storage control then searches for index before presenting final status.

**COMPARISON EQUAL** channel end/device end and status modifier are presented to the channel.

**COMPARISON UNEQUAL** channel end/device end are presented to the channel. If multiple track bit is on, search continues on next track if command is reissued. Multiple track will not cross cylinder boundary.

**CCW COUNT  $\geq$  FOUR BYTES** search is completed when four bytes are received by storage control; command is terminated with channel end/device end (and status modifier if comparison was equal).

**CCW COUNT  $<$  FOUR BYTES** comparison of storage bytes and head address continues until CCW count is decremented to zero. Channel end/device end are presented to channel when 2835 count reaches zero. Status modifier is presented with channel end and device end if search was satisfied on short field.

**BUS OUT PARITY** detected; channel end, device end, and unit check presented in ending status.

**Note:** *Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH ID EQUAL

Command Code	Data Address	Flags	Count
Binary <b>0011 0001</b> Hex <b>31</b> Multitrack <b>1011 0001</b> Hex <b>B1</b>	Specifies main storage location of a five-byte record identifier (CC HH R).	Used at the discretion of the programmer.	Five

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH ID EQUAL** compares the main storage ID and the count area ID. ID to be compared is next ID on the track.

**INITIAL STATUS** normally zero.

**COMPARISON EQUAL** channel end/device end/status modifier presented to the channel.

**COMPARISON UNEQUAL** channel end/device end presented to the channel.

**CCW COUNT  $\geq$  FIVE** first five bytes used.

**CHANNEL END/DEVICE END** presented to terminate command.

**STATUS MODIFIER** presented if comparison was equal.

**CCW COUNT  $<$  FIVE** comparison of main storage and track data continues until **CCW** count is zero.

**CHANNEL END/DEVICE END** presented to channel when ID and correction code bytes are read and checked.

**STATUS MODIFIER** presented if search is satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index. Sense data will indicate no record found.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or head number reaches 8 (end of cylinder).

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of end of cylinder. Sense data will indicate end of cylinder.

**Note:** *Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH ID HIGH

Command Code	Data Address	Flags	Count
Binary <b>0101 0001</b> Hex <b>51</b> Multitrack <b>1101 0001</b> Hex <b>D1</b>	Specifies main storage location of a five byte record identifier (CC HH R).	Used at the discretion of the programmer.	Five

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH ID HIGH** compares the main storage ID and the count area ID. ID to be compared is next ID on the track.

**INITIAL STATUS** normally zero.

**TRACK ID HIGHER** than main storage ID: channel end/device end/status modifier presented to the channel.

**TRACK ID EQUAL OR LOWER** than main storage ID: channel end/device end presented to the channel.

**CCW COUNT  $\geq$  FIVE** first five bytes used.

**CHANNEL END/DEVICE END** presented to terminate command.

**STATUS MODIFIER** presented if comparison was high.

**CCW COUNT  $<$  FIVE** comparison of main storage and track data continues until CCW count is zero.

**CHANNEL END/DEVICE END** presented to channel when ID and correction code bytes are read and checked.

**STATUS MODIFIER** presented if search is satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or end of cylinder is reached.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of end of cylinder.

**Note:** *Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH ID EQUAL OR HIGH

Command Code	Data Address	Flags	Count
Binary <b>0111 0001</b> Hex <b>71</b> Multitrack <b>1111 0001</b> Hex <b>F1</b>	Specifies main storage location of a five byte record identifier (CC HH R).	Used at the discretion of the programmer.	Five

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH ID EQUAL OR HIGH** compares the main storage ID and the count area ID. ID to be compared is next ID on the track.

**INITIAL STATUS** normally zero.

**TRACK ID EQUAL OR HIGHER** than main storage ID: channel end/device end/status modifier presented to the channel.

**TRACK ID LOWER** than main storage ID: channel end/device end presented to the channel.

**CCW COUNT  $\geq$  FIVE** first five bytes used.

**CHANNEL END/DEVICE END** presented to terminate command.

**STATUS MODIFIER** presented if comparison was equal or high.

**CCW COUNT  $<$  FIVE** comparison of main storage and track data continues until CCW count is zero.

**CHANNEL END/DEVICE END** presented to channel when ID and correction code bytes are read and checked.

**STATUS MODIFIER** presented if search is satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or head number reaches 8 (end of cylinder).

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of end of cylinder.

**Note:** *Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH KEY EQUAL

Command Code	Data Address	Flags	Count
Binary <b>0010 1001</b> Hex <b>29</b> Multitrack <b>1010 1001</b> Hex <b>A9</b>	Specifies main storage locations to which key is compared.	Used at the discretion of the programmer.	Equal to key length.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH KEY EQUAL** compares main storage key to key area read from track. Key to be compared is next key on track.

*Note: When command is chained from Search ID or Read Count, key compared is in same record as ID or count.*

**INITIAL STATUS** normally zero.

**COMPARISON EQUAL** channel end, device end, and status modifier presented to the channel.

**COMPARISON UNEQUAL** channel end/device end presented to the channel.

**CCW COUNT  $\geq$  KL** search operation completed when key area is read.

**CHANNEL END/DEVICE END** terminates command.

**STATUS MODIFIER** presented if comparison was equal.

**CCW COUNT  $<$  KL** track and main storage data comparison continues until CCW count is zero.

**CHANNEL END/DEVICE END** presented after key area and the following correction code bytes are read and checked.

**STATUS MODIFIER** presented if search was satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or head number reaches 8 (end of cylinder).

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of end of cylinder.

**COMMAND EXECUTION** on a record with zero KL does not set a status modifier. If followed by a chained read data command; the data area read is that of the next record.

*Note: Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH KEY HIGH

Command Code	Data Address	Flags	Count
Binary <b>0100 1001</b> Hex <b>49</b> Multitrack <b>1100 1001</b> Hex <b>C9</b>	Specifies main storage locations to which key is compared.	Used at the discretion of the programmer.	Equal to key length.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH KEY HIGH** compares main storage key to key area read from track. Key to be compared is next key on track.

*Note: When command is chained from search ID or read count, key compared is in same record as ID or count.*

**INITIAL STATUS** normally zero.

**TRACK KEY** higher than main storage key: channel end/device end/status modifier presented to the channel.

**TRACK KEY EQUAL OR HIGHER** than main storage key: channel end/device end presented to the channel.

**CCW COUNT  $\geq$  KL** search operation completed when key area is read.

**CHANNEL END/DEVICE END** terminates command.

**STATUS MODIFIER** presented if comparison was high.

**CCW COUNT  $<$  KL** track and main storage data comparison continues until CCW count is zero.

**CHANNEL END/DEVICE END** presented after key area and the following correction code bytes are read and checked.

**STATUS MODIFIER** presented if search was satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or head number reaches 8 (end of cylinder).

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of end of cylinder.

**COMMAND EXECUTION** on a record with zero KL does not set a status modifier. If followed by a chained read data command, the data area read is that of the next record.

*Note: Successive searches without an intervening operation on a data field may produce unpredictable results.*

## SEARCH KEY EQUAL OR HIGH

Command Code	Data Address	Flags	Count
Binary <b>0110 1001</b> Hex <b>69</b> Multitrack <b>1110 1001</b> Hex <b>E9</b>	Specifies main storage locations to which key is compared.	Used at the discretion of the programmer.	Equal to key length.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**SEARCH KEY EQUAL OR HIGH** compares main storage key to key area read from track. Key to be compared is next key on track (excluding R0).

*Note: When command is chained from Search ID or Read Count, key compared is in same record as ID or count. Search key equal bypasses R0 unless chained from Search ID command which successfully searched R0 ID.*

**INITIAL STATUS** normally zero.

**TRACK KEY EQUAL OR HIGHER** than main storage key: channel end/device end/status modifier presented to the channel.

**TRACK KEY LOWER** than main storage key: channel end/device end presented to the channel.

**CCW COUNT  $\geq$  KL** search operation completed when key area is read.

**CHANNEL END/DEVICE END** terminates command.

**STATUS MODIFIER** presented if comparison was equal or high.

**CCW COUNT  $<$  KL** track and main storage data comparison continues until **CCW** count is zero.

**CHANNEL END/DEVICE END** presented after key area and the following correction code bytes are read and checked.

**STATUS MODIFIER** presented if search was satisfied on the short field.

**MULTI-TRACK NOT USED** search is confined to one track; search is repeated (as long as channel repeats command) until search condition is satisfied or two index points are detected.

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**MULTI-TRACK USED** causes search to continue (as long as channel repeats command); head number automatically increments at index until search condition is satisfied or head number reaches 8 (end of cylinder).

**CHANNEL END/DEVICE END/UNIT CHECK** presented to channel upon detection of second index.

**COMMAND EXECUTION** on a record with zero KL does not set a status modifier. If followed by a chained read data command, the data area is that of the next record.

*Note: Successive searches without an intervening operation on a data field may produce unpredictable results.*



## READ HOME ADDRESS

Command Code	Data Address	Flags	Count
Binary <b>0001 1010</b> Hex <b>1A</b> Multitrack <b>1001 1010</b> Hex <b>9A</b>	Specifies main storage location where data from 2835 is to be stored.	Used at the discretion of the programmer.	Five

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ HOME ADDRESS** transfers the five bytes of data to main storage.

Bytes 0, 1, and 3 are always zero.

Byte 2 = cylinder address.

Byte 4 = head address.

**INITIAL STATUS** normally zero.

**TRACK DATA** is *not* transferred.

**DATA TRANSFER COMPLETE** causes 2305 to search for index.

**CHANNEL END/DEVICE END** presented when index is detected.

**Note:** *No home address written on track. Command maintained for compatibility with other direct access storage devices.*

## READ COUNT

Command Code	Data Address	Flags	Count
Binary <b>0001 0010</b> Hex <b>12</b> Multitrack <b>1001 0010</b> Hex <b>92</b>	Specifies main storage locations where first byte of count data is to be transferred.	Used at the discretion of the programmer.	Eight

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ COUNT** transfers the eight bytes (CC HH R KL DL DL) of the next count area encountered on the track (excluding R0) from disk storage to main storage.

**INITIAL STATUS** normally zero.

**DATA VALIDITY** is verified by correction code bytes following the count area.

**SERVICE OVERRUN** , if detected, is signaled at end of count area.

**COMMAND OVERRUN** , if detected, is signaled immediately.

**CHANNEL END/DEVICE END/UNIT CHECK** presented for either of the above.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**CHANNEL END/DEVICE END** are signaled to the channel at completion of the correction code check.

## READ RECORD ZERO

Command Code	Data Address	Flags	Count
Binary <b>0001 0110</b> Hex <b>16</b> Multitrack <b>1001 0110</b> Hex <b>96</b>	Specifies main storage location where first byte of R0 count data is to be transferred.	Used at the discretion of the programmer.	Eight + key length + data length of R0.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ R0** transfers count, key and data areas of R0 from disk storage to the channel.

**INITIAL STATUS** normally zero.

**STORAGE CONTROL** searches for index, clocks through gap 1.

**DATA TRANSFER** of the R0 count area is initiated by storage control.

**DATA VALIDITY** is verified by correction code bytes following each area.

**SERVICE OVERRUN** , if detected, is signaled at end of area in which error occurred.

**COMMAND OVERRUN** , if detected, is signaled immediately.

**CHANNEL END/DEVICE END/UNIT CHECK** status presented with either of above.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**COMMAND EXECUTION** is accomplished immediately if read R0 is chained from a search home address or read home address command; the storage control will not search for index in these cases.

**CHANNEL END/DEVICE END** are presented to the channel at completion of the correction code check of the data area.

## READ DATA

Command Code	Data Address	Flags	Count
Binary <b>0000 0110</b> Hex <b>06</b> Multitrack <b>1000 0110</b> Hex <b>86</b>	Specifies main storage location where first byte of data is to be transferred.	Used at the discretion of the programmer.	Data length of record read.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ DATA** transfers the data area of a record from disk storage to main storage. The data transferred is:

1. Data area read by Search ID or Search key command from which read command is chained.
2. Data area read by Read Count command from which command is chained.
3. Data area of record following next count area on the track (excluding R0).

**INITIAL STATUS** normally zero.

**DATA VALIDITY** is verified by correction code bytes following each area.

**SERVICE OVERRUN** , if detected, is signaled at end of data area.

**COMMAND OVERRUN** , if detected, is signaled immediately.

**CHANNEL END/DEVICE END/UNIT CHECK** presented with both of above.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**CHANNEL END/DEVICE END** are presented to the channel at completion of the correction code check of the data area.

R  
E  
A  
D

## READ KEY AND DATA

Command Code	Data Address	Flags	Count
Binary <b>0000 1110</b> Hex <b>0E</b> Multitrack <b>1000 1110</b> Hex <b>8E</b>	Specifies main storage location where first byte of key data is to be transferred.	Used at the discretion of the programmer.	Key length + data length of record read.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ KEY AND DATA** transfers key and data areas of a record from disk storage to main storage. The key and data areas are:

1. Key and data area of record read by Search ID or Read Count command from which read key and data is chained.
2. Key and data areas of record following next count area on the track (excluding R0).

**INITIAL STATUS** normally zero.

**DATA VALIDITY** is verified by correction code bytes following each area.

**SERVICE OVERRUN** , if detected, is signaled at end of area in which error was detected.

**COMMAND OVERRUN** , if detected, is signaled immediately.

**CHANNEL END/DEVICE END/UNIT CHECK** presented with either of above.

**KEY LENGTH = ZERO** command operates as a Read Data command.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**CHANNEL END/DEVICE END** are presented to the channel at completion of the correction code check of the data area.

## READ COUNT, KEY and DATA

Command Code	Data Address	Flags	Count
Binary <b>0001 1110</b> Hex <b>1E</b> Multitrack <b>1001 1110</b> Hex <b>9E</b>	Specifies main storage location where first byte of count data is to be transferred.	Used at the discretion of the programmer.	Eight + key length + data length of record read.

**CHAINING AND SPECIAL REQUIREMENTS** See the Multiple Track Operations section and the additional programming considerations when using multiple track commands.

**READ COUNT, KEY AND DATA** transfers the next record encountered on the track from disk storage to main storage (excluding R0).

**INITIAL STATUS** normally zero.

**DATA VALIDITY** is verified by correction code bytes following each area.

**SERVICE OVERRUN** , if detected, is signaled at end of area in which error occurred.

**COMMAND OVERRUN** , if detected, is signaled immediately.

**CHANNEL END/DEVICE END/UNIT CHECK** presented with either of above.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**CHANNEL END/DEVICE END** are signaled to the channel at completion of the correction code check of the data area.

## READ INITIAL PROGRAM LOAD

Command Code	Data Address	Flags	Count
Binary 0000 0010 Hex 02	Specifies main storage location where first byte of data is to be transferred.	Used at the discretion of the programmer.	Data length of first record after R0 on track 0 of cylinder 0.

**CHAINING AND SPECIAL REQUIREMENTS** must not be preceded by a set file mask in the same chain. See Note.

**READ INITIAL PROGRAM LOAD** causes storage control to seek to cylinder 0, head 0 of selected drive and search for index.

**INITIAL STATUS** normally zero.

**DATA AREA** read, after index is detected, is the first record after R0.

**COMMAND INITIATION** is *normally* accomplished by designating the direct access storage device to be used as the loading unit and initiating an IPL procedure.

**DATA VALIDITY** is verified by correction code bytes following the data area.

**CHANNEL END/DEVICE END** are signaled to the channel at completion of the correction code check.

**PARITY BIT** is added to each byte prior to transferring byte to the channel.

**Note:** *The Read IPL command should be the first CCW in a chain. If Read IPL is not the first, results are unpredictable.*

## READ SECTOR

Command Code	Data Address	Flags	Count
Binary <b>0010 0010</b> Hex <b>22</b>	Specifies main storage location where sector number is to be stored.	Used at the discretion of the programmer.	One

**READ SECTOR** transfers one byte of data from storage control to main storage.

**INITIAL STATUS** normally zero.

**BYTE TRANSFERRED** contains sector number of the last count field accessed by the current CCW chain.

*Note: If a Read, Write, or Search CCW has not occurred since the beginning of the chain, or the last Set Sector CCW, the value of the sector number returned will be unpredictable.*

**COMMAND EXECUTION** resets orientation information in the storage control.

**CHANNEL END/DEVICE END** presented after sector number is transferred.



## SENSE

Command Code	Data Address	Flags	Count
Binary 0000 0100 Hex 04	Specifies storage location where bytes are to be transferred.	Used at the discretion of the programmer.	Twenty-four

**SENSE** transfers up to twenty-four bytes of sense information from the storage control to the channel.

**INITIAL STATUS** normally zero.

### DESCRIBES:

#### UNIT CHECK STATUS

**CURRENT STATUS** of the device that performed operation, and

**SYSTEM ERROR RECOVERY** information.

**UNIT CHECK** should always be followed by a Sense command, whether or not sense information is used; otherwise expected future interrupts may not occur and some I/O access paths may be unavailable.

**CHANNEL END/DEVICE END** presented after sense bytes are transferred.

See "Sense Bytes" for a description of the sense information pertaining to 2835/2305 operations.

READ BUFFERED LOG

Command Code	Data Address	Flags	Count
Binary 0010 0100 Hex 24	Specifies main storage location of first error byte or usage information.	Used at the discretion of the programmer.	128

**READ BUFFERED LOG** transfers up to 128 bytes of usage or error information from storage control to the channel.

**INITIAL STATUS** normally zero.

**USAGE/OVERRUN/ERROR INFORMATION** is generated by the overflow of the respective counters and makes up 15th and 16th bytes of buffered log. A command with an error log full condition, to any device address, causes a unit check. The subsequent sense data (sense byte 2, bit 0) indicates that a read buffered log command should be issued.

**CHANNEL END/DEVICE END** presented after data transfer.

**CCW COUNT < 128** specified number of bytes transferred, but log information not reset.

**CCW COUNT ≥ 128** 128 bytes transferred and log information reset to zero.

**Note:** Refer to Appendix C for log format.



## DEVICE RELEASE

Command Code	Data Address	Flags	Count
Binary <b>1001 0100</b> Hex <b>94</b>	Specifies main storage location where sense bytes are to be transferred.	Used at the discretion of the programmer.	Twenty-four

**CHAINING AND SPECIAL REQUIREMENTS** must be first command in a chain.

**INITIAL STATUS** normally zero.

**SENSE INFORMATION** (up to twenty-four bytes) is transferred to the channel.

**NORMAL BUSY** condition causes the command to be rejected with a busy indication.

**ABNORMAL FILE** status conditions (file unsafe, off-line, etc.) do not halt command execution.

**CHANNEL END/DEVICE END** presented after sense bytes are transferred.

**UNIT CHECK** , causing command rejection, is presented if:

Two channel switch feature is not installed in storage control.  
Command is not first CCW in the chain.

*Note: The drive will remain reserved until all of its associated logical device addresses are released. Every logical device address which was issued a Device Reserve command must receive a Device Release command before the module is released. Thus, if logical addresses 1 and 4 were each issued a Device Reserve, addresses 0-7 will not be released until both addresses 1 and 4 are issued a Device Release command.*

*When a release is complete, any interrogation from the other channel which had received Device Busy status during the reserve condition is presented Device End status via polling sequences.*

## DEVICE RESERVE

Command Code	Data Address	Flags	Count
Binary <b>1011 0100</b> Hex <b>B4</b>	Specifies main storage location where sense bytes are to be transferred.	Used at the discretion of the programmer.	Twenty-four

**CHAINING AND SPECIAL REQUIREMENTS** must be first command in a chain.

**DEVICE RESERVE** command reserves the addressed logical device to the channel issuing the command, and also reserves all other logical devices on the same drive module. A reserve to any logical address is an effective reserve to all eight logical addresses associated with a drive module.

**INITIAL STATUS** normally zero.

**RESERVATION MAINTAINED** until either a Device Release command or a system reset is performed by the channel.

**SENSE INFORMATION** (up to twenty-four bytes) is transferred to the channel.

**NORMAL BUSY CONDITIONS** causes the command to be rejected with a busy indication.

**ABNORMAL FILE STATUS** conditions (file unsafe, off-line, etc.) do not halt command execution.

**UNIT CHECK** , causing command rejection, is presented if:

Two channel switch feature is not installed in storage control.  
Command is not first CCW in the chain.

**CHANNEL END/DEVICE END** presented after sense byte transfer if all other logical device addresses on the drive can be implicitly reserved.

**CHANNEL END** presented after sense byte transfer if operations initiated by the other interface are in process. (New operations initiated by the other interface will receive a busy indication.)

**DEVICE END** presented when all outstanding operations on the other interface have been completed. The drive module (8 addresses) is reserved.

**Note:** *All addresses for the drive must be available before device can be reserved. See Unit Selection and Device Addressing.*

S  
E  
N  
S  
E

## READ DIAGNOSTIC STATUS 1

Command Code	Data Address	Flags	Count
Binary <b>0100 0100</b> Hex <b>44</b>	Specifies main storage location where data accumulated during prior diagnostic load or diagnostic write is to be stored.	Used at the discretion of the programmer.	16 or 512

**CHAINING AND SPECIAL REQUIREMENTS** See Note.

**READ DIAGNOSTIC STATUS 1** may perform either of two functions:

### COMMAND FOLLOWS A DIAGNOSTIC WRITE COMMAND:

**ERROR CODE MESSAGE** (up to 16 bytes) transferred from storage control buffer to main storage.

**CCW COUNT** should specify 16 bytes. CCW count < 16 causes command reject with unit check in ending status.

**CHANNEL END/DEVICE END** presented after transfer.

### COMMAND FOLLOWS A DIAGNOSTIC LOAD COMMAND:

**DIAGNOSTIC TEST** (512 bytes) transferred from storage control buffer to main storage.

**CCW COUNT** should specify 512 bytes. CCW count < 512 causes command reject with unit check in ending status.

**CHANNEL END/DEVICE END** presented after transfer.

**INITIAL STATUS** normally zero.

**DIAGNOSTIC LOAD/DIAGNOSTIC WRITE** must precede the Read Diagnostic Status 1 command; otherwise sixteen bytes of data are transferred from storage control buffer area. In this case, the first byte is hex FF and the remaining bytes are undefined.

**CHANNEL END/DEVICE END** are presented after data transfer.

*Note: This command will not execute with 2835 microcode for basic and two channel switch (P/N 2254580) or four channel switch RPQ (P/N 2254581). These two microcode loads are used only for attachment to the 3081 Model D. Using this command with these microcodes will cause a unit check with sense byte 7=C1 to be returned.*

## WRITE HOME ADDRESS

Command Code	Data Address	Flags	Count
Binary <b>0001 1001</b> Hex <b>19</b>	Used during alternate track sparing for transferring defective track information from system to control storage.	Used at the discretion of the programmer.	Five

**CHAINING AND SPECIAL REQUIREMENTS** must be preceded by a set file mask permitting Write Home Address commands. See Notes 1, 2, 3, and 4.

**WRITE HOME ADDRESS** is used primarily for compatibility with other direct access storage devices.

**INITIAL STATUS** normally zero.

**COMMAND EXECUTION** causes no data to be written to the disk.

**STORAGE CONTROL** requests five bytes (FCC HH)\* and orients on index. See Note 5.

**CHANNEL END/DEVICE END** presented upon index detection.

### Notes:

1. All references in Notes 2 and 4 to a CCHH = 00600000 are for a 2835/2305 Model 2. Use a CCHH = 00300000 for a Model 1.
2. Before the alternate track spare is invoked, the user must ensure that it has not been previously allocated. This can be checked by issuing a Seek with an argument of CCHH = 00600000 followed by a Read R0 and then comparing the ID bytes of the count field with the seek address of the track. If the compare is equal, the alternate track spare address has not been allocated. If the compare is not equal, the spare has been invoked and allocated for the defective track.
3. If the alternate track spare has been previously invoked, any subsequent attempt to write the home address with the defective track indicator on (flag byte, bit 6 = 1) causes a unit check and command reject. Sense byte 7 contains hex '85' (second alternate track spare attempted).
4. The following sequence resets the address pointer in control storage: issue a Write Home Address command with the defective track indicator on and with a CCHH = 00600000. This is followed by a Write R0 command containing the same CCHH = 00600000. The alternate track spare can be invoked by a Write Home Address command with the defective track indicator on and a CCHH = defective track address. This is followed by a Write R0 command with a CCHH = defective address. Control storage now contains the defective track address. The 2835 substitutes the alternate track spare any time a seek is made to the defective track. The second Write R0 command records the defective track address in the ID bytes of the count field, which are read and stored in control storage with each initial control logic load.
5. If a count less than five bytes is received and the flag byte is on, then command reject and unit check are posted in ending status. If a count less than five bytes is received and the flag byte is off, then the storage control unit orients to index and posts channel end and device end.

\* The flag byte (F) must be zero except when allocating the alternate spare. Bit 6 of the flag byte invokes the alternate spare and assigns it to the address of the CCHH bytes.

## WRITE R0

Command Code	Data Address	Flags	Count
Binary <b>0001 0101</b> Hex <b>15</b>	Specifies main storage location of R0 count, key and data bytes.	Used at the discretion of the programmer.	Specifies total number of bytes in R0 count, key, and data areas.

### CHAINING AND SPECIAL REQUIREMENTS

1. Must be preceded by a set file mask permitting Write R0 commands.
2. Must be chained from a successful Search HA, Read HA, or Write HA.

**WRITE R0** causes specified data in main storage to be written on selected drive.

**INITIAL STATUS** normally zero.

**COUNT AREA** is made up of the first eight bytes from main storage.

*Note: The flag byte is generated by storage control; the remaining data is written in the key and data areas as specified by the KL and DL bytes in the count area.*

**CORRECTION CODE BYTES** are written by storage control at the end of each record area.

**CCW COUNT FIELD** specifies the number of bytes ( $8 + KL + DL$ ) to be transferred from main storage to the storage module.

$CCW\ COUNT < 8 + KL + DL$  storage control writes 0's in remainder of record.

**CHANNEL END/DEVICE END** is signaled after correction code bytes are written for the data area.

## ERASE

Command Code	Data Address	Flags	Count
Binary <b>0001 0001</b> Hex <b>11</b>	Specifies main storage location where count, key, and data areas of the record are located.	Used at the discretion of the programmer.	Specifies number of bytes in count, key, and data areas of the record.

**CHAINING AND SPECIAL REQUIREMENTS** must be chained from either Write R0, Write CKD, Write Special CKD, \*Search ID Equal, or \*Search Key Equal.

**ERASES** one or more records from the track, beginning with oriented record and continuing through all remaining records until the index is detected on that track.

**INITIAL STATUS** normally zero.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented in initial status.

**DATA TRANSFERRED FROM CHANNEL** but not written on disk.

**TRACK** is erased to index.

\* Search commands must compare equal on all bytes of the recorded field.



## WRITE COUNT, KEY, and DATA

Command Code	Data Address	Flags	Count
Binary <b>0001 1101</b> Hex <b>1D</b>	Specifies main storage location where count, key and data bytes of record are located.	Used at the discretion of the programmer.	Specifies total number of bytes in count, key, and data areas.

**CHAINING AND SPECIAL REQUIREMENTS** must be chained from either Write R0, Write CKD, \*Search ID Equal, or \*Search Key Equal.

**WRITE COUNT, KEY, AND DATA** causes specified data in main storage to be written on selected drive.

**INITIAL STATUS** normally zero.

**COUNT AREA** is made up of the first eight bytes from main storage.

**FLAG BYTE** is generated by storage control; the remaining data is written in the key and data areas as specified by the KL and DL bytes in the count area.

**CORRECTION CODE BYTES** are written by storage control at the end of each record area.

**CCW COUNT FIELD** specifies number of bytes ( $8 + KL + DL$ ) to be transferred from main storage to the drive.

**CCW COUNT**  $< 8 + KL + DL$  storage control writes 0's in the remainder of the record.

**READ DATA/READ KEY AND DATA** may be inserted between Search CCW and Write CKD CCW.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented in initial status.

**CHANNEL END/DEVICE END** are signaled to the channel after data area correction code bytes are written for the data area.

\* Search commands must compare equal on all bytes of the recorded field.

## WRITE SPECIAL COUNT, KEY, and DATA

Command Code	Data Address	Flags	Count
Binary 0000 0001 Hex 01	Specifies main storage location where count, key and data areas of the record are located.	Used at the discretion of the programmer.	Specifies number of bytes in the count, key, and data areas of the record segment.

**CHAINING AND SPECIAL REQUIREMENTS** must be chained from a Write R0, Write CKD, \*Search ID Equal or \*Search Key Equal command.

**WRITE SPECIAL COUNT, KEY, AND DATA** formats a segment of an overflow record; last segment is written by a normal Write CKD command.

**INITIAL STATUS** normally zero.

**COUNT AREA** is made up of the first eight bytes from main storage.

**FLAG BYTE** contains a 1 in bit position 1; generated and written by the storage control, this bit indicates that another part of the record is located on the next track.

**CORRECTION CODE BYTES** are written by storage control at the end of each record area.

**CCW COUNT FIELD** specifies number of bytes (8 + KL + DL) to be transferred from the main storage to the drive.

**CCW COUNT < 8 + KL + DL** storage control writes 0's in the remainder of the record.

**READ DATA/READ KEY AND DATA** may be inserted between Search CCW and Write Special CKD CCW.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented in initial status.

**CHANNEL END/DEVICE END** are signaled to the channel after data area correction code bytes are written for the data area.

\* Search commands must compare equal on all bytes of the recorded field.

## WRITE DATA

Command Code	Data Address	Flags	Count
Binary <b>0000 0101</b> Hex <b>05</b>	Specifies main storage location of data used to update record.	Used at the discretion of the programmer.	Data length of selected record.

**CHAINING AND SPECIAL REQUIREMENTS** must be chained from a \*Search ID Equal or \*Search Key Equal command.

**INITIAL STATUS** normally zero.

**COMMAND EXECUTION** causes specified data in main storage to be written in data area of selected record.

### NUMBER OF BYTES WRITTEN

1. is specified in the count field of the write data CCW.
2. may be less than data length specified in formatted record.

**CCW COUNT < COUNT AREA DL** storage control writes 0's in remaining data area, writes ECC bytes and presents channel end/device end to channel.

**CCW COUNT ≥ COUNT AREA DL** channel end/device end presented when DL byte count = zero.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented in initial status.

\* Search commands must compare equal on all bytes of the recorded field.

## WRITE KEY and DATA

Command Code	Data Address	Flags	Count
Binary 0000 1101 Hex 0D	Specifies main storage location of data to be used to update record.	Used at the discretion of the programmer.	Sum of (key length + data length) of specified record.

**CHAINING AND SPECIAL REQUIREMENTS** must be chained from a \*Search ID Equal command.

**WRITE KEY AND DATA** is used for record updating after track formatting.

**INITIAL STATUS** normally zero.

**COMMAND EXECUTION** causes data from main storage to be written in key and data area of selected record.

### NUMBER OF BYTES WRITTEN

1. is specified in the count field of the Write Key and Data CCW.
2. may be less than key and data length specified in formatted record.

**CCW COUNT < KL/DL BYTE COUNT** storage control writes 0's in the remaining areas, writes ECC bytes, and presents channel end/device end to channel.

**CCW COUNT ≥ KL/DL BYTE COUNT** channel end/device end presented when KL/DL byte count = 0.

**CHAINING REQUIREMENTS** must be met; otherwise channel end, device end, and unit check are presented in initial status.

**CHANNEL END/DEVICE END** presented after ECC bytes have been written for the data area.

\* Search command must compare equal on all bytes of the recorded field.

## Channel Programs

The following channel programs are typical examples of how CCW's are arranged to format, read, and write records on the 2835/2305 facility. The examples do not include the processor program which would be used to initiate the channel program. All examples within this section are for Format 0.

Unless otherwise noted, all numbers used are hexadecimal.

Example 1: Format track 2A on head 4 with record zero and records R1, R2, and R3 for customer records. Assume R0 has a key length of zero and a data length of eight bytes;

and, assume R1, R2, and R3 have a key length of 6 bytes and a data length of 03E8 (1000 bytes).

The channel program used is:

Seek  
Set File Mask  
Set Sector  
Write Record Zero  
Write CKD  
Write CKD  
Write CKD

### SEEK

Command Code	Data Address	Flags	Count
Binary <b>0000 0111</b> Hex <b>07</b>	C C H H 03E8 = 00 00 00 2A 00 04	Chain command	0006

Comments: The seek command is used to access the desired cylinder and to select the proper head. All seek commands transfer six bytes of data from main storage to the storage control. (Thus the byte count of six.) The first 3 bytes of the seek address are always 0's, the cylinder number (2A) is specified in the fourth byte, byte five is zero and byte six specifies the desired head (00 04 at 03EC and 03ED).

## SET FILE MASK

Command Code	Data Address	Flags	Count
Binary 0001 1111 Hex 1F	03EE = C0	Chain command	0001

Comments: The Set File Mask command is used to specify the types of operations that can be performed in this channel program. The mask byte in this case (1100 000 at address 03EE) permits all write and seek commands. The mask is reset to zero at the end of each chain of commands.

## SET SECTOR

Command Code	Data Address	Flags	Count
Binary 0010 0011 Hex 23	1390 = 00	Chain command	0001

Comments: Execution of a Set Sector command with an argument of zero orients the track to index. During the time that the 2835 is waiting for index, the channel is available to perform other operations on other devices.

## WRITE R0

Command Code	Data Address	Flags	Count
Binary 0001 0101 Hex 15	<div>C C H H R KL DL DL</div> <div>07D0 = 00 2A 00 04 00 00 00 08</div> <div>07D8 = 00 00 00 00 00 00 00 00</div>	Chain command	0010

Comments: The Write R0 command writes a count area and a data area whose length is specified in the DL bytes of the count area. In this example, the data address is at 07D0 and a byte count of sixteen is specified.

Since the key length specified is zero, address 07D5 is coded 00 and no key area is written. The data length is eight bytes so addresses 07D6 and 07D7 are coded 00 08, and the data in the following eight main storage locations is written in the data area.

Note that the byte count in the Write R0 command is sixteen and the 2835 requested sixteen bytes (eight for the count area and eight for the data area). Therefore, no incorrect length error is generated.

## WRITE CKD (3 CCW's Shown)

Command Code	Data Address	Flags	Count
Binary <b>0001 1101</b> Hex <b>1D</b>	R1 C C H H R KL DL DL 0BB8 = 00 2A 00 04 01 06 03 E8	Chain command, SLI	0008
	R2 0FA0 = 00 2A 00 04 02 06 03 E8	Chain command, SLI	0008
	R3 1388 = 00 2A 00 04 03 06 03 E8	SLI	0008

Comments: Execution of the Write CKD commands cause a count area, key area (if the key length specified is not zero), and a data area whose length is dependent upon the value specified in the DL bytes of the count area, to be written on the disk.

The main storage locations specified in the data address are coded with the cylinder number, head number, record number, key length, and data length of each record. Since the key length specified is six, a key area six bytes long will be created. The data length specified is 03E8 (1000 bytes). Although the CCW byte count is only eight, and the channel byte count will go to zero after eight bytes have been written, the 2835 is committed to writing a key area six bytes long and a data area 1000 bytes long. Therefore, the 2835 inserts 0's in the applicable byte positions on the track until the 2835 byte count reaches zero.

The difference in the channel byte count and the 2835 byte count will cause an incorrect length indication; therefore the SLI flag is on in the CCW's.

In this example, six bytes of 0's will be recorded in the key area followed by 10 or 16 error correction code bytes, a gap, 1000 bytes of 0's and 10 or 16 more error correction code bytes. At a later time, data can be recorded in the key and data areas with the following CCW sequence.

Set Sector  
Search ID Equal  
TIC \*-8  
Write Key and Data  
Search ID Equal (R2)  
etc.

Example 2: Update Frank Smith's payroll record. Assumed:

1. The disk is organized by key areas.
2. Each key area contains an employee number.
3. Frank Smith's employee number 656151.
4. This employee number is located on track 0C head 04.
5. Key areas are 6 bytes long and data areas 64 (100<sub>10</sub>) bytes long.

The channel program used is:

Seek

Search Key Equal

TIC \*-8

Write Data

## SEEK

Command Code	Data Address	Flags	Count
Binary <b>0000 0111</b> Hex <b>07</b>	C C H H 03E8 = 00 00 00 0C 00 04	Chain command	0006

Comment: As explained in example 1, the Seek command transfers the track address to the storage control and selects the specified head.

## SEARCH KEY EQUAL

Command Code	Data Address	Flags	Count
Binary <b>0010 1001</b> Hex <b>29</b>	(employee no.) 07D0 = F6 F5 F6 F1 F5 F1	Chain command	0006

Comments: After locating the proper cylinder and track, it is necessary to find Frank Smith's record. Since the disk is organized by keys, a Search Key Equal command is executed. Execution of this command causes the 2835 to search the key field of the next record encountered on the track. If the key is not equal to Frank Smith's employee number, (main storage locations 07D0 to 07D5) the 2835 signals channel end and device end to the channel and the TIC command (back to search key equal) is executed. Subsequent key areas are searched until Frank Smith's record is found. The 2835 then signals channel end, device end, and status modifier to the channel. The status modifier bit in the ending status byte causes the channel to skip the next command (TIC) and execute the Write Data command.



## TRANSFER IN CHANNEL (TIC \*-8)

Comment: TIC \*-8 branches back to the last command (Search Key Equal) if the compare is not successful.

## WRITE DATA

Command Code	Data Address	Flags	Count
Binary <b>0000 0101</b> Hex <b>05</b>	(data to update record) 0BB8 = XX XX XX to 0C1B	none	0064

Comments: The Write Data command transfers the data to update Frank Smith's payroll record from the main storage locations 0BB8 to 0C1B to the disk.

**Note:** *If Frank Smith's payroll record had not been on track 0C head 04, the program would loop between the search key equal and TIC until every key on the track had been searched. The 2835 would then signal unit check to the channel. A subsequent Sense command would indicate no record found.*

The data just written could be verified by chaining the following CCW's to the Write Data command:

- Read Sector (store sector address)
- Set Sector (locate sector)
- Search Key Equal (locate record)
- TIC \*-8
- Read Data (verify data)

Example 3: Find and read Joe Brown's insurance policy number. Assume:

1. The disk is organized by ID — no keys.
2. Joe Brown's employee number is 12320.
3. The data length of each record is 00AA (170 bytes).
4. His policy number is in the data area.
5. The data set begins on cylinder 0A, track 00.

Using the record capacity chart in Appendix A, it is known that forty 170 byte records without keys can be written on a 2305 Model 2 track. Since the disk is organized by IDs

(Joe Brown's = 12320), the track and record location can be determined by dividing the ID by the number of records per track. In this case:

$$\frac{12320}{40} = 308 \quad \text{Note: Add 1 to the remainder to establish the address of the specific record.}$$

Thus, Joe Brown's ID is 308 tracks from the beginning of the data set.

The CC HH R for the Seek command is then determined by converting the 308 tracks to cylinders and adding the results to the beginning of the data set.

	Cylinder	Head	Record	C	C	H	H	R
Starting Address:	10	00	0	00	0A	00	00	00
Displacement: *	38	04	1	00	26	00	04	01
Result:	48	04	1	00	30	00	04	01

\*Determined by dividing 308 by 8.

The channel program used is:

```
Seek
Search ID Equal
TIC *-8
Read Data
```

## SEEK

Command Code	Data Address	Flags	Count
Binary 0000 0111 Hex 07	C C H H 03E8 = 00 00 00 30 00 04	Chain Command	0006

Comment: The Seek command is executed to select cylinder 30 (decimal 48) and head 04.

## SEARCH ID EQUAL

Command Code	Data Address	Flags	Count
Binary <b>0011 0001</b> Hex <b>31</b>	C C H H R 05DC = 00 30 00 04 01	Chain command	0005

Comments: The Search ID Equal command causes the first ID encountered on the track to be compared with Joe Brown's ID. All unequal comparisons of IDs cause the 2835 to signal channel end/device end to the channel, and the TIC command (back to the Search ID Equal) is executed. When an equal comparison is encountered (ID of record 1) the 2835 signals channel end, device end, and status modifier to the channel. Status modifier causes the next command (TIC) to be skipped and the Read Data command is executed.

If the Search ID Equal is not satisfied and index is passed twice, unit check is sent in the status byte. A subsequent Sense command would indicate no record found. The course of action would then be determined by the system error recovery procedures.

## TRANSFER IN CHANNEL (TIC \*-8)

Comment: TIC \*-8 branches back to the last command (Search ID Equal).

## READ DATA

Command Code	Data Address	Flags	Count
Binary <b>0000 0110</b> Hex <b>06</b>	(insurance policy no.) 0BB8 = XX XX XX to 0C61	None	00AA

Comment: Execution of the Read Data command causes the data area, containing Joe Brown's insurance policy number, to be read into main storage location 0BB8 to 0C61.

## Standard Features

### Search Operations

The 2835 incorporates two modes of searching, oriented and unoriented. An oriented search results from a command that is chained from a previous write or read. The Read Sector and Read Buffered Log commands are excluded from the set of read commands that establish orientation. The search begins immediately from the end of the last field processed by the prior command. The search ends upon reaching index, and is not continued past the index point (except on multi-track operations, where the search will continue to the next track(s) until the search is satisfied, or end of cylinder is detected).

An unoriented search results from a command that is not chained from a previous write or read. The search begins on the next address mark upon receipt of the command and initial status acceptance by the channel. The search ends after index is detected twice.

In both modes, oriented and unoriented, if the target record is not located, a no record found indication is posted.

If a full track search is desired, it should be unoriented (un-chained), or chained exclusively from a Read Home Address or Read R0 command.

### Multiple Track (MT) Operation

On all search, and most read commands, the storage control can automatically select the next sequentially numbered head on the disk drive under control of bit 0 of the command code. If bit 0 is a 1 and data transfer of the command has not been initiated, the next sequentially numbered head is selected at index. This eliminates the need for seek head commands in a chain of read or search commands.

Channel end, device end, and unit check is signaled to the channel if the head switching operation crosses a file protected boundary, or exceeds the limits of the pseudo-cylinder.

Use discretion when using the multiple track commands. To retrieve a record N, perform the following two steps to ensure locating the required record:

1. Orient with a single track command (read or search with the MT bit 0 off) to a position prior to the desired record N. If the track location of record N is known, orient on that specific track; if the location of record N is not known, orient on head 0 of the desired cylinder.
2. Use a multiple track command (MT bit 0 on) to read or search for record N on that track, or for a subsequent track of the desired cylinder.

If during a multi-track search operation the desired record is on the first track searched, but the search commences after that record is passed, the head number is advanced to the next track without comparing on the desired record. To avoid this condition a single track Read Home Address or Read R0 should be placed before the search to ensure that the search commences at R0 or R1 of the track. (See Figure 5.)

Multi-track operations are not used on Read IPL, Read Sector, or Read Diagnostic Status 1 commands.

### Record Overflow

The record overflow function provides a means of processing logical records that exceed the capacity of a track. When using overflow records, the limiting factor to the size of the record is the cylinder boundary.

#### *Formatting Overflow Records*

The portion of an overflow record that is written on (or read from) one track is called a record segment. Each segment contains a count field, key field (optional), and a data field. The key and data lengths specified in the KL and DL bytes of the count field pertain to that segment, not the entire overflow record. Since only the key field of the first segment has significance, overflow records are usually formatted without key fields (KL=0).

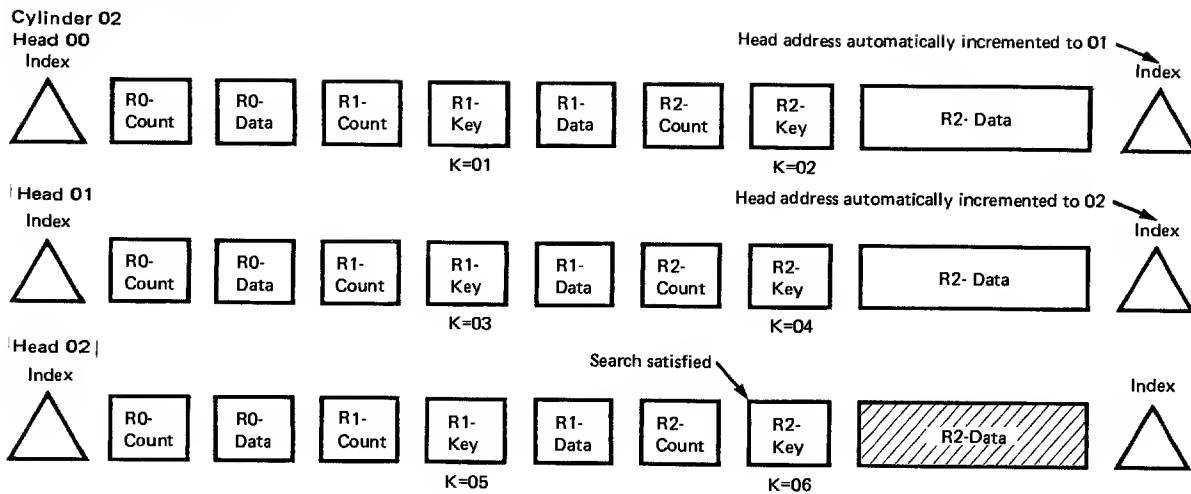
Write Special Count, Key, and Data commands are used to format all segments of an overflow record except the last segment. As shown in Figure 6, the last segment is formatted with a normal Write Count, Key, and Data command.

Write Special CKD commands cause a 1 to be written in bit position one of the flag byte of the record segment being written. This bit identifies the record as an overflow segment and indicates to subsequent commands processing the record, that the logical record continues on the following track.

No internally generated head switching is associated with formatting overflow records. All head seeking must be done by the formatting program (Figure 6). During read and write update operations, head switching will not occur in violation of the file mask or past the end of the cylinder.

All segments of an overflow record except the first must be written immediately following R0, and all segments except the last must be the last physical record on their respective tracks.

## Multi-Track Operation



Channel program using multiple track search.

Object: Update John Doe's payroll record.  
 Assume: The disk is organized by keys, and the physical address of the record is unknown.  
 Set File Mask (allow write and seek commands).  
 Seek (cylinder 02, head 00).  
 Read Home Address (make sure all records are read).  
 Search Key Equal (MT bit on, argument = 06).  
 TIC \*-8  
 Write Data (updates shaded area).

Figure 5. Multiple Track Operation

## Processing Overflow Records

The following commands may be used to read or update previously formatted overflow records.

- Read Count, Key, and Data.
- Read Key and Data.
- Read Data.
- Write Key and Data.
- Write Data.

When any of the above commands are used to process an overflow record, the operation does not terminate at the end of a record segment if the segment is flagged with bit 1 (on) in the flag byte. Instead, the head address is incremented by 1 at index, and the operation continues in the data field of record one on the next track. If this record segment is also flagged with bit 1 (on) in the flag byte, the operation continues on the next track. When a segment is found that is not flagged, the operation terminates at the end of the data field. The net effect of this procedure is that the data fields of all the record segments appear as a single logical data field.

If a data check or bus out parity error occurs, unit check is signaled at the end of the associated area.

If a write operation was in progress when the data check or bus out parity error occurred, unit check is signaled at the end of the record segment.

If the CCW count is less than number of bytes in the logical record, the operation continues to the end of the logical record before presenting ending status.

Spacing over overflow records does not occur automatically. The channel program must be written so that the entire logical record is spaced over, not just the first segment. For example, in the sequence:

```
Search ID (first segment)
TIC *-8
Read CKD (multi-track)
```

the read CKD does *not* read the next logical record on the cylinder. It reads starting at the count field of the second segment of the overflow record.

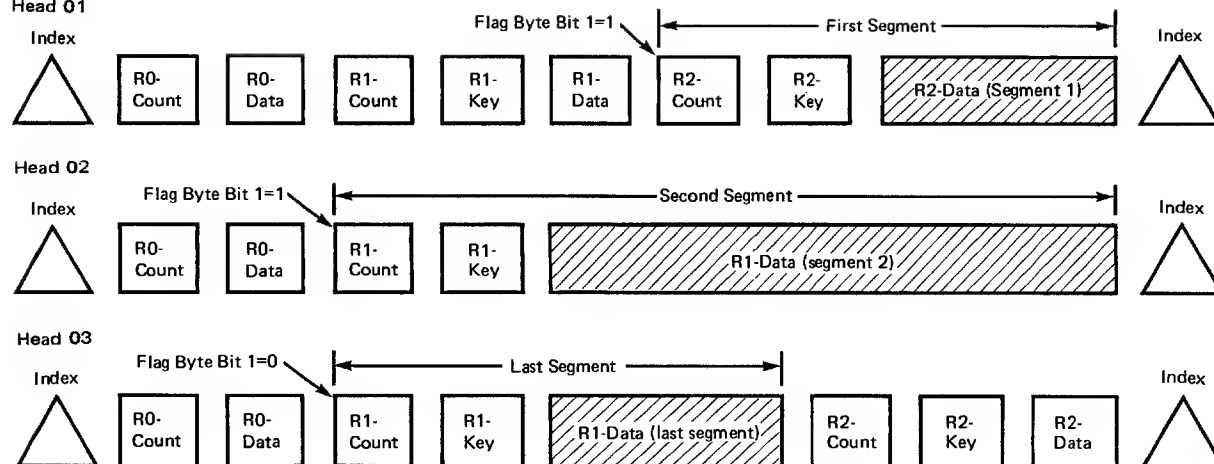
The sequence:

```
Search ID (first segment)
TIC *-8
Read Key and data (skip and SLI flags on)
Read CKD (multi-track)
```

reads the count, key, and data of the next logical record.

## Overflow Record

Cylinder 02  
Head 01



Typical channel programs for formatting, updating, and reading overflow records.

### Formatting:

Set sector  
Search ID R1 (track 1)  
TIC\*—8  
Write special CKD (segment 1)  
Seek head (next track)  
Search ID R0 (track 2)  
TIC\*—8  
Write special CKD (segment 2)  
Seek head (next track)  
Search ID R0 (track 3)  
TIC\*—8  
Write CKD (last segment)

### Updating:

Set sector  
Search ID R2 (segment 1)  
TIC\*—8  
Write data (updates shaded areas)

### Reading:

Set Sector  
Search ID R2 (segment 1)  
TIC\*—8  
Read data (reads shaded areas)

Figure 6. Record Overflow

Multiple track operations should not be confused with overflow record operations. When processing overflow records, head switching occurs regardless of whether the MT bit is on or off.

Truncation of a write on the base segment or any subsequent segment of an overflow record will result in write padding the remainder of the physical field being processed. Write padding will not propagate into the remaining logical record (if any). The user must pad or erase the remaining logical record, if required.

## End-of-File

An end-of-file record is used to define the end of a logical group of records. An end-of-file record is written by executing a Write Count, Key, and Data command with the DL bytes in the count area set to zero. Execution of a Write CKD with a data length of zero causes the storage control to write a data area consisting of one byte of 0's (2 bytes on a model 1) followed by the error correction code bytes (Figure 7).

The KL portion of the count area can be either zero or non-zero. If KL equals zero, the end-of-file record contains the contents of the count area and data area only. If the key length is not zero, the key area is written as specified by the KL byte.

Detection of a data length of zero causes unit exception status to be generated. No data from the data area is transferred to the channel. A Read R0, Read CKD, or Read KD will transfer the key area (if any) to the channel.

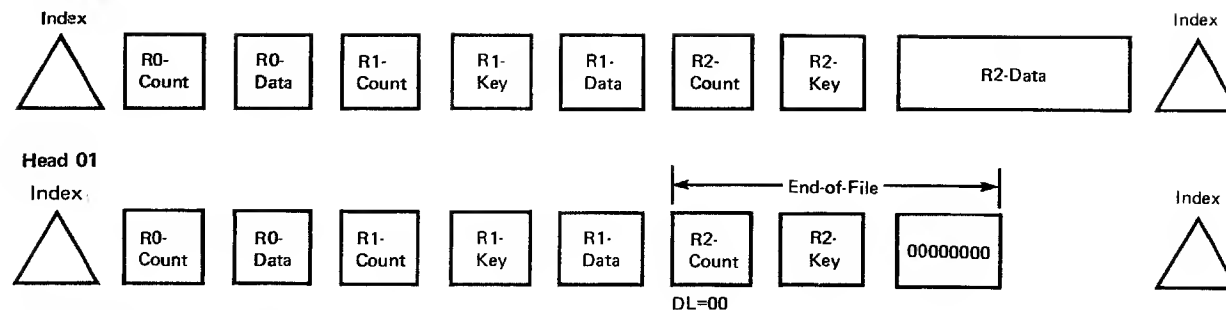
The unit exception is generated during execution of Read IPL, Read R0, Read CKD, Read KD, Read Data, Write KD, and Write Data commands.

**Note:** The end-of-file indicator should not be embedded within a logical overflow record. If the indicator is embedded, results are unpredictable.

## End-Of-File

### Cylinder 02

#### Track 00



Channel program with end-of-file record.

```
Set File Mask (allow seek and write)
Seek (cylinder 02, head 00)
Write Home Address
Write R0
Write CKD R1
Write CKD R2
Seek Head (cylinder 02, head 01)
Write Home Address
Write R0
Write CKD R1
Write CKD R2 (data length = 00)
```

Figure 7. End-Of-File

## Command Retry

Command retry is a channel/storage control procedure that causes an improperly executed command in a channel program to be automatically retried. The re-execution does not cause an interrupt and programmed error recovery procedures are not required. Command retry is used:

1. To recover from correctable data errors (error burst of 11 bits or less) that occur during a read or search operation on a count or key area.

During a read or search operation, the count or key area read from the disk is placed in a buffer in the control unit. When a correctable error occurs, the control unit corrects the data in the buffer and re-issues the command that detected the error. During re-orientation to the record, the channel is free. When the failing command is re-executed, the corrected data from the buffer is used instead of the data actually on the track.

When a correctable error occurs in the data field of a record, correction information is sent to the channel in the sense data.

2. When an uncorrectable data error is detected, the failing command is reissued. If the error does not recur, the CCW chain continues. If the error recurs, the storage control retries the operation again.
3. Command retry is also used to recover from command overrun and service overrun.

## Multiple Requesting

The multiple requesting function provides the capability for record request queueing within the 2835/2305 facility. This queueing is accomplished by allowing multiple set sector commands to be issued to a single disk module.

The function is implemented by associating up to eight logical (system) device addresses with a single physical module. This permits the channel to issue a set sector command to one logical device, disconnect on channel end status, and then issue a set sector command to another logical device. The arguments transferred by the set sector command are stored in the storage control. Whenever the storage control is not executing a command and is not otherwise busy, it monitors the angular position counters in the attached disk modules. When a counter compares equal with one of the stored arguments for that module, the storage control raises request-in and, when polled, presents device end status for the appropriate logical device.

To properly complete a chain when the channel reconnects, the storage control must store the arguments of Set File Mask and Seek commands issued previously in the same chain. If multiple requests are pending against a module, the proper head may no longer be selected when the channel reconnects to complete the chain. If this is the case, the storage control ensures that the proper head is reselected prior to raising request-in.

### ***Rotational Position Sensing***

Rotational position sensing enables the channel to issue a seek to an angular track position. It permits channel disconnection during most of the rotational delay period and thus contributes to increased channel utilization.

The storage control implements rotational position sensing by dividing each track into equal angular segments. There are 90 segments per track in the 2305 Model 1 and 180 segments per track in the 2305 Model 2.

The channel obtains the angular position of the given record by chaining a Read Sector command to a read, write, or search command that has operated on the record. The channel locates to a particular angular position by executing the Set Sector command.

The sector location of a record is determined by the length of all records that are ahead of it and its sequential position on the track. The sector location can be calculated with the following formulas.

For 2305 Model 1: Sector Number =  $0.00595X$

Where X = a function of the number of bytes from index to the beginning of the desired record. The sector value for record zero is 1. For records beyond record zero, the value of X is:

$$X = 234 + \sum_{n=0}^{N-1} \left( 430 + C_n + 2 \left\lceil \frac{KL_n + 1}{2} \right\rceil + 2 \left\lceil \frac{DL_n + 1}{2} \right\rceil \right)$$

Where:  $n$  = consecutive record number.  
 $N$  = consecutive record number of desired record.  
 $KL_n$  = Key length of record  $n$ .  
 $DL_n$  = Data length of record  $n$ .  
 $C_n = 0$  if  $KL_n = 0$ .  
 $C_n = 202$  if  $KL_n \neq 0$ .

For 2305 Model 2: Sector Number =  $0.0119 X$

$$\text{Where } X = 118 + \sum_{n=0}^{N-1} (198 + C_n + KL_n + DL_n)$$

Where:  $n$  = consecutive record number.  
 $N$  = consecutive record number of desired record.  
 $KL_n$  = Key length of record  $n$ .  
 $DL_n$  = Data length of record  $n$ .  
 $C_n = 0$  if  $KL_n = 0$ .  
 $C_n = 91$  if  $KL_n \neq 0$ .

An example of rotational position sensing is shown in Figure 8.



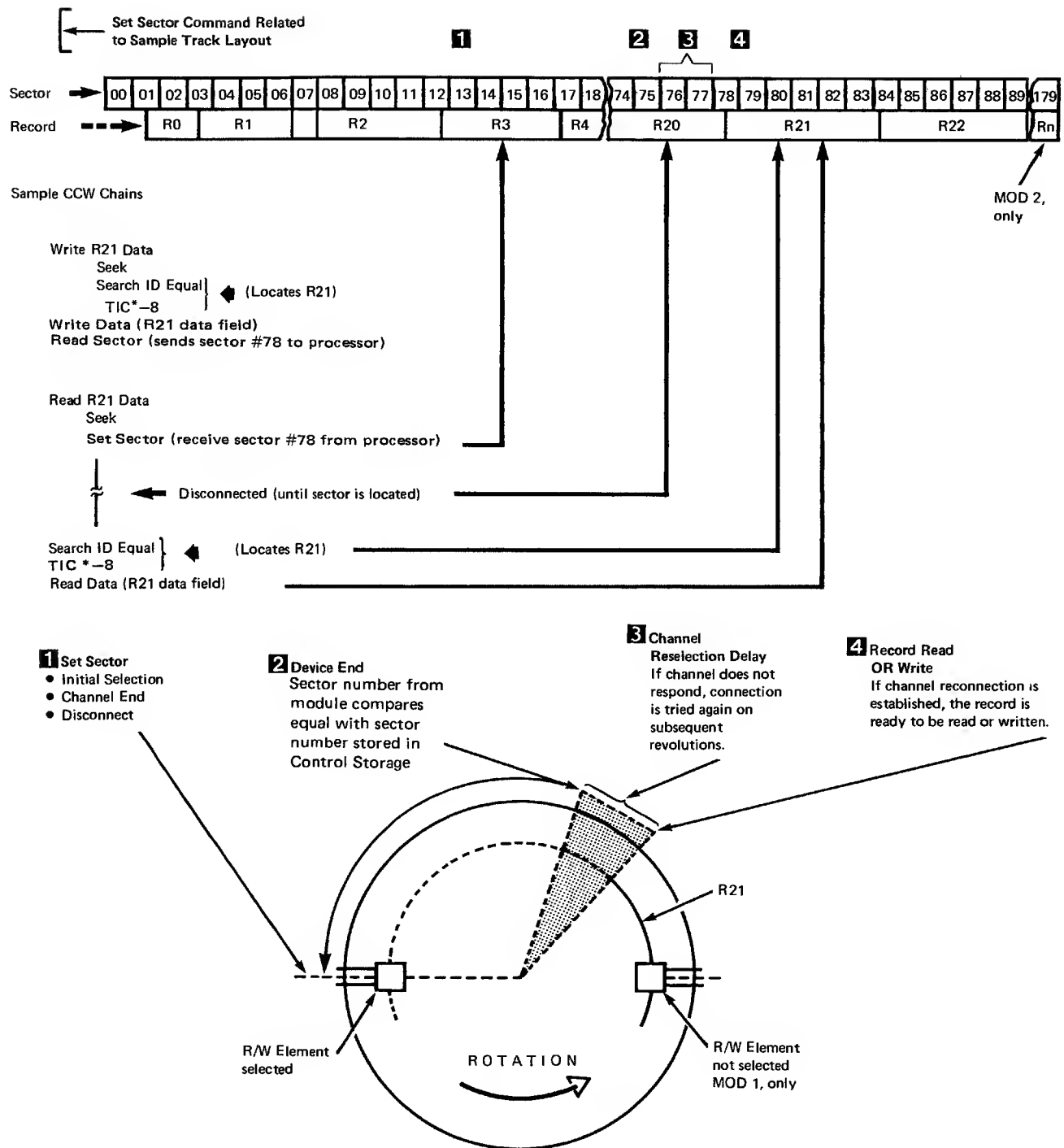


Figure 8. Rotational Position Sensing

## File Protection

The file protection scheme used with direct access storage devices depends on a file mask byte (see Set File Mask command) properly set to restrict certain operations. One such operation is the prevention of seeks beyond the cylinder boundary. Since the 2305 has a read/write head for each addressable track, the entire module could be considered one huge cylinder. For the purpose of file protection and programming compatibility, the 2305 is divided into pseudo-cylinders of eight tracks each. The tracks are grouped as follows:

Cylinders	Track
1	0-7
2	8-15
3	16-23
etc.	etc.
96	760-767

The 2305 Model 1 contains 48 pseudo-cylinders; the 2305 Model 2 contains 96 pseudo-cylinders. A pseudo-cylinder is treated by the 2305 in the same manner that a cylinder is treated by other devices. That is, a multiple track search operates only on one pseudo-cylinder before returning an end-of-cylinder status indication. Also, the file mask restricts seek commands within the pseudo-cylinder boundary.

## Storage Control Diagnostics

To provide maximum facility availability, the 2835 can execute diagnostic tests on a drive concurrent with normal system operations on the remaining drives. This mode of operation allows the customer engineer to diagnose and repair most drive failures while the facility continues to operate other attached drives. The 2835 provides a transient block of 512 bytes (128 words) of control storage to allow temporary residence for a specific diagnostic test.

The transient area is loaded by the system under control of the On-Line Test Executive Program (OLTEP). A special command (diagnostic write), loads a selected test into control storage and instructs the storage control to execute the test. This loading and execution can also be initiated from the CE panel.

After the test, error message information or test results are transferred from the 2835 to main storage by the diagnostic sense command. If the CE panel is used, the test results are displayed in the CE panel indicators.

### *Usage/Overrun/Error Counters*

The storage control maintains a set of counters for each drive. These counters accumulate total bytes read from the files, missing address marks, and all detected overruns. The counters are initialized with values that cause the byte counter to overflow if the predicted error rates are not exceeded. When a drive counter overflows, the contents of all counters for that drive are transferred to the buffered log, and the counters are reinitialized. The transfer is accomplished by the read buffered log command. Execution of this command causes the current contents of the counters to form the 15th and 16th entries in the log area (see Appendix C).

### *Buffered Log*

The buffered log accumulates log entries of one or more of the following types:

1. Catastrophic storage control failures that result in a disconnect-in channel sequence.
2. Storage control data errors.
3. Control logic load file errors recovered by internal retry.
4. Usage/overrun/error counters that are logged when an overflow occurs.

The buffered log is transferred to the channel when a read buffered log command is executed. (See Appendix C for log entry formats.)

### *Disconnect In*

Disconnect in is activated by the storage control when a catastrophic error prevents normal control logic termination of the sequence. It can be activated only when the storage control is connected to the channel.

The channel responds to disconnect in by performing a selective reset and causing an I/O interruption with an interface control check indication.

### *Special Microcodes for 2835 Model 2 Attachment to 3081 Model D Processors*

When attaching a 2305/2835 Model 2 to support a 3081-D system, the concurrent microdiagnostics and On-Line Test (OLT) maintenance capability is lost to all channel ports of the 2835 Storage Control.

This limitation will affect other types of processors when the “3081 type” functional microcode is resident in the 2835 Storage Control. For example, with only one channel port connected to a 3081 channel, the concurrent inline microdiagnostic capability is lost on all channels attached to the 2835, regardless of the other processor type(s).

For maintenance purposes, the customer engineer will be required to take the entire 2305/2835 facility offline, with all the drums attached, to run microdiagnostics or OLT.

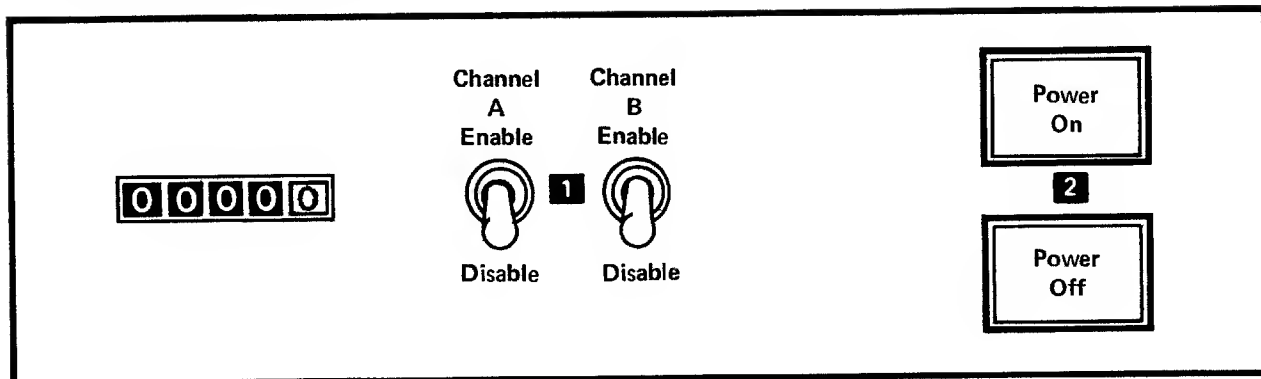
When the maintenance is completed, the “3081 type” functional microcode will be reloaded by the CE, prior to returning the storage facility to the customer for use.

In addition, the dynamic alternate spare capability has been removed from the automatic internal microcode loading routine. However, it can be assigned by an initializing utility, such as ICK DSF, and the Write HA command.

The On-Line Test (OLT) limitation exists because the diagnostic commands (53, 73, and 44) cannot be executed with the special microcode. A unit check, with intervention required (Sense byte 0 = 40) and inline contention (Sense byte 7 = C1) results when these diagnostic commands are decoded.

## Operator Controls and Indicators

2835 Storage Control Panel



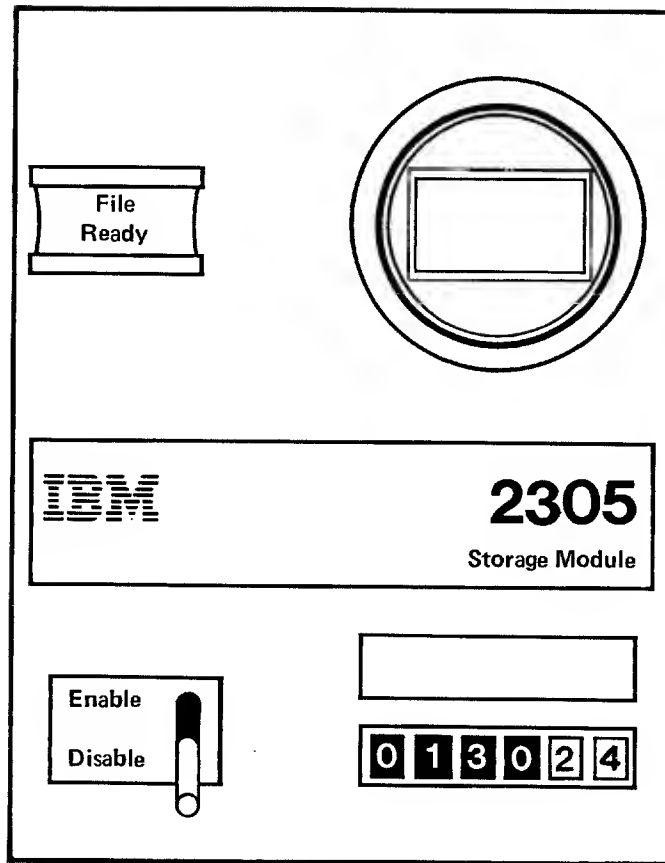
**1** Toggle switch that must be in the enable position before the 2835 storage control is available to the channel. If two-channel switch feature is installed, a separate switch is provided for each channel.

**2** **Power Off:** A momentary pushbutton that can be used to remove ac power from the 2305 facility.

If system power is on when the pushbutton is pressed, ac power is removed from the 2305 facility. If system power is later turned off, then on, ac power is reapplied to the 2305 facility; operation of the power on pushbutton is not required.

**Power On:** A momentary pushbutton that can be used to reverse the effect of the power off switch. If system power is on, and the power off switch is pressed to remove ac power from the 2305 facility, then pressing the power on switch will restore ac power to the 2305 facility.

Whenever system power is brought up, ac power is applied to the 2305 facility, regardless of what was previously done to the two pushbuttons.



**Ready Indicator:** When illuminated, this light indicates that the module identified by the number on the indicator is ready. After power is applied, a warm-up period of about 5 minutes is required before the module is ready for processing.

**Enable/Disable:** Enables or disables communication between the associated device and control unit. The processor must be in the stop or wait state for the transfer of the switch from one position to the other to be effective.

**Usage Meter:** Records actual process time.

## Special Features

### Two Channel Switch

The two channel switch special feature provides the ability for the 2835 Storage Control to be shared by two channels. The channels may be attached to the same or different central processing units. Individual storage modules attached to the storage control may be reserved for the exclusive use of either of the channels. Channel switching and device reservation are controlled by the channel program. Two special commands are associated with two channel switch operation, device reserve and device release (see Channel Commands).

#### *Device Reservation*

Reservation of a 2305 module is accomplished by executing the device reserve command. (Reservation of any logical device address causes reservation of the physical module-eight addresses.) Once a module is reserved by a particular channel, it remains reserved to that channel until that channel releases it by executing a device release command for each logical address that was previously issued a reserve command.

When a module is busy with channel A (for any reason) or reserved to channel A, any command from channel B addressed to that module is rejected with a busy indication in the initial status byte. After the reservation has been terminated, the busy indication causes the 2835 to attempt to present channel B with a status byte containing the device end. The device address byte presented with this ending status byte is the same as that associated with the busy status byte.

#### *Channel Selection Switch*

Channel selection is determined by the “setting” of a three position switch in the 2835. When the switch is in neutral, the 2835 can be selected by either channel. The channel A position indicates that the storage control has been selected by channel A, and the channel B position indicates that the storage control has been selected by channel B.

Once the 2835 has been selected by either channel, it is reserved to that channel until ending status is presented. (See Ending Status and Multiple Requesting.) The channel selection switch will then return to neutral unless:

1. Chaining is indicated and device end is included in the status.
2. A contingent connection is established (see Contingent Connection).

When a channel connection is maintained as a result of a contingent connection, the storage control will not respond to polling by the channel, except to present stacked status or control unit end for the exposure under contingent connection. When the contingent connection is terminated, the channel selection switch returns to neutral.

The length of time the storage control is connected to a channel determines how the selection switch responds to a channel initiated sequence from the other channel. The channel/storage control connection is classified as either instantaneous or long.

*Note: In the following descriptions, the conditions of both channels are reversible.*

**Instantaneous Connections:** At the moment the channel selection switch connects to a channel, the connection is considered to be instantaneous; the state of a channel connection changes from instantaneous to long if:

1. The channel is initiating or resuming a channel program.
2. The channel indicates chaining on device end during a polling sequence.

If a channel/storage control connection is caused by a condition other than 1 or 2, the state of the connection remains instantaneous until the operation is complete and the selection switch returns to neutral.

During an instantaneous connection on channel A, the selection switch does not respond to a channel initiated selection sequence on channel B until the instantaneous connection is terminated. At the termination of the instantaneous connection, the selection switch either remains connected to channel A (if the connection state changed from instantaneous to long), or returns to neutral. If the switch returns to neutral from channel A during a channel initiated selection sequence on channel B, the switch immediately connects to channel B. If the connection state changed from instantaneous to long on channel A, the response of the selection switch is as described in Long Connection.

**Long Connection:** During a long connection on channel A, the storage control responds with a short control unit busy sequence (busy and status modifier) to a channel initiated selection sequence on channel B.

Whenever the short control unit busy sequence occurs, the storage control attempts to present control unit end status to channel B after the selection switch returns to neutral. The address byte associated with this status is the address of the lowest numerical device address which is not implicitly connected to either interface.

The pending control unit end status does not cause the storage control to appear busy to channel A as long as the selection switch is not actually connected to channel B.

#### *Device Status*

**Multi-Tagged Status:** Multi-tagged status conditions cause device end status to be generated for each of the attached channels. Multi-tagged status is the result of a device end status generated by a transition from not-ready to ready, i.e., power on or Enable/Disable switched to Enable. The multi-tagged device end status must be accepted by each channel to clear the status. Acceptance of the device end status by one channel does not clear the status for the other channel.

**Tagged Status:** Tagged status is associated with a particular channel and is available only to that channel. Tagged status remains pending until accepted by the channel identified by the tag.

#### *Addressing*

The base address (four high-order bits) of the storage control on one channel is independent of the base address on the other channel. However, the four low-order address bits for any attached device must be the same on both channels.

#### *Resets*

A system reset can be initiated by either channel at any time. A system reset: (1) resets all reservations and status conditions stored in the storage control for the resetting channel, (2) terminates all block multiplex command chains in progress on the resetting channel A, (3) resets all device interrupts not associated with the other channel. Reservations, status, and device interrupts for, as well as block multiplex chains in progress on, the other channel is not affected. If a channel initiates a system reset while the selection switch is connected to the other channel, a machine reset is performed when the selection switch goes to neutral. A selective reset has no effect on device reservations or status.

# Error Correction Function and Related Tables

## Error Condition Table

The error condition table (Figure 9) identifies the primary sense bits (bytes 0-2) that are posted by the 2835, and maps each into a specific recovery action to be invoked by the system.

## Error Correction Function

The 2835 Storage Control posts the data check and correctable sense bits if a correctable data error is detected in the data area during the processing of a basic command.

Correctable data errors in count and key areas are corrected internally by utilizing command retry. Such cases are not posted with the correctable and data check sense bits and do not cause a system interrupt.

If the correctable and data check sense bits are included in the sense information, sense bytes 18 through 22 provide the error pattern and displacement as shown in the following illustration:

Sense byte	18	19	20	21	22
Contents	Displacement		Error Pattern		

The displacement is used to align the error pattern with the erroneous bytes in main storage. The displacement specifies the number of bytes the error pattern must be shifted relative to the last byte transferred.

The error pattern is used to correct the erroneous data in main storage. Error correction is performed by aligning the error pattern bytes with the bytes specified by the displacement and exclusively ORing the error pattern with the main storage data.

## Special Cases

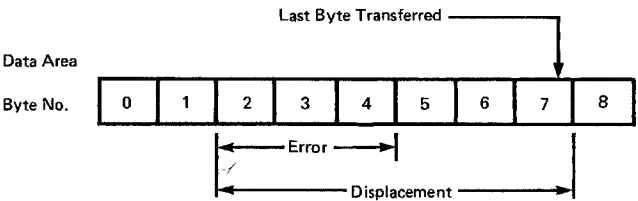
If the displacement is equal to 0, the error burst occurred in data bytes which were not transferred to main storage due to truncation. In this case, the storage control sets the three error pattern bytes to 0 before transferring the sense information. The error recovery procedure bypasses the exclusive OR operation and continues to the next step as specified in the recovery action table.

If the skip bit is on in the failing CCW, data was not transferred to main storage. In this case, the error correction function should be bypassed, the error recovery procedure should be continued as specified in the recovery action table.

## Error Correction Example for Model 2

To clarify the correction procedure, the following example is given:

Assume a data area is nine bytes in length, only eight bytes were transferred to main storage because of a short count in the CCW, and the error pattern has a displacement of 6.



Assume the error affected bytes 2, 3, and 4 as follows:

2	3	4
-----XXXXX-----	X-----	
Error Burst		

where: (—) corresponds to a correct bit  
(X) corresponds to an incorrect bit

The condition illustrated above generates sense bytes 18, 22 as follows:

Sense Bytes 18 and 19  
00000000 00000110  
Sense Bytes 20, 21, and 22  
00000011 11100000 10000000

Error correction is accomplished by aligning the error pattern with main storage bytes 2, 3, and 4 and exclusively ORing the error pattern with the main storage bytes.

**Note:** For Model 1, the data is correctable only if the error occurs on one path. Because every other byte of the preceding example is written on path 0 and path 1, respectively, the error burst affects bytes 2, 4, and 6 rather than 2, 3, and 4. The exclusive OR in main storage would be with bytes 2, 4, and 6.

## Construction of Restart CCW

If operation incomplete (byte 1, bit 7) of the sense information is set, a logical operation was in progress and data transfer had been initiated when an error or unusual condition occurred. The error recovery procedure corrects the unusual condition and continues the operation that was in



progress from the point of interruption to the normal ending point. Sense byte 3 provides the restart command code.

The restart CCW is constructed with the following procedure:

1. The command code byte is provided in sense byte 3.
2. The data address is that of the interrupted CCW, plus the count of that CCW, minus the residual count.
3. The flags (except for PCI) are those of the interrupted CCW.
4. The count is the residual count. If the residual count is 0, then a count of 1 may be used, and the CCW address should point to a byte of 00 (hex) in main storage.

The programmer should be aware that a second operation incomplete might occur while executing the restart CCW. In this case, a new restart CCW may be generated from the old restart CCW, using the procedure described, but care must be used in not destroying the old restart CCW before generating the new one.

### Recovery Action Table

The recovery action table (Figure 10) specifies an error correction function (ECF) as a necessary step in the recovery procedure. The section entitled "Error Correction Function" provides the ECF algorithm and related instructions that are required.

In the error correction and recovery action examples, ICCWA is used for Interrupted Channel Command Word Address, and ICCWA-8 for Interrupted Channel Word Address minus eight.

Error Condition Table			
Sense		Condition	Action
Byte	Bit		
		Channel Control Check	5
		Interface Control Check	5
		Channel Data Check	6
		Chaining Check	6
		Unit Check	12
1	0	Permanent Error	1
0	3	Equipment Check	6
0	2	Bus Out	3
0	1	Intervention Required	4
0	0	Command Reject	2
1	4	No Record Found	2
0	5	Overrun	6
1	1	Invalid Track Format	2
0	4	Data Check	7
1	2	End of Cylinder	8
1	5	File Protect	9
2	0	Buffered Log Full	11
		Program Check	2
		Protection Check	2
		Unit Exception	2
		Incorrect Length	2
<b>Note:</b> All of the indicators defined above must be checked in sequence until the one caused by the error condition is found.			

Figure 9. Error Condition Table

Recovery Action Table	
Action Number	Action
1	Provide an operator message and exit to permanent error.
2	Exit with a permanent error.
3	(A) Retry the chain of commands from the last restart point once. (B) On the second occurrence of the error condition do Action 1.
4	(A) Retry the chain of commands from the last restart point once. (B) On the second occurrence of the error condition, provide an Operator message. Retry the chain of commands when the device becomes ready.
5	(A) If CCH (Channel Check Handler) is incorporated in the system, and if the No Retry bit in the ERPIB is not on, do Action 6. (B) Otherwise, do Action 1.
6	(A) Retry the chain of commands from the last restart point ten times. (B) If the error condition persists after ten retries, do Action 1.
7	(A) If the correctable bit (Byte 2, bit 1) is not on, do Action 6. (B) If the correctable bit is on, perform ECF. (C) If Operation Incomplete is on, do Action 10. (D) If the failing CCW was the last CCW in the chain, exit with no error. (E) If the failing CCW was not the last CCW in the chain, execute the following CCW chain: Seek (CCHH provided in sense bytes 4-7) Set File Mask (Same as original) Set Sector ('S' provided in sense byte 9) Search ID equal (CCHHR provided in sense bytes 4-8) TIC *-8 NOP (Space over error record) TIC (ICCWA)
8	(A) Update the users seek argument to the next cylinder and head 0. (B) If Operation Incomplete is on, do Action 10(B). (C) Execute the following CCW chain: Seek (Argument from Step A) Set File Mask (Same as original) Set Sector 0 TIC (ICCWA-8)
9	(A) If the interrupted CCW is a seek command, execute the following CCW chain: Seek (Users argument) Set File Mask (Same as original) TIC (ICCWA) (B) If the interrupt was caused by a multi-track operation, increment the users seek argument by 1. (C) If Operation Incomplete is on, do Action 10(B). (D) Continue the operation by executing the following CCW chain: Seek (Argument from Step B) Set File Mask (Same as original) TIC (ICCWA-8)
10	(A) Increments the seek argument (CCHH as provided in Sense bytes 4-7) by 1. (B) Construct the restart CCW. Sense Byte 3 provides the restart command code. (C) Complete the interrupted operation and continue the user's chain (if appropriate) by executing the following CCW chain: Seek (Argument from Step A) Set File Mask (Same as original) Search ID Equal (Record 1) Restart CCW TIC (ICCWA)
11	Transfer contents of Buffered Log to main storage.
12	Perform a sense command and continue with the checks as indicated in Error Condition Table.

Figure 10. Recovery Action Table

## Appendix A: Record Track Capacities

2305 Model 1 Capacity							
Without Keys							
Bytes Per Record		Records Per				Byte Capacity Per Module	
Minimum	Maximum	Track	Cylinder	Module	Facility	Minimum	Maximum
23	36	31	248	11904	23808	273792	428544
37	52	30	240	11520	23040	426240	599040
53	70	29	232	11136	22272	590208	779520
71	88	28	224	10752	21504	763392	946176
89	106	27	216	10368	20736	922752	1099008
107	128	26	208	9984	19968	1068288	1277952
129	150	25	200	9600	19200	1238400	1440000
151	174	24	192	6216	18432	1391616	1603584
175	200	23	184	8832	17664	1545600	1766400
201	230	22	176	8448	16896	1698048	1943040
231	260	21	168	8064	16128	1862784	2096640
261	296	20	160	7680	15360	2004480	2273280
297	334	19	152	7296	14592	2166912	2436864
335	376	18	144	6912	13824	2315520	2598912
377	424	17	136	6528	13056	2461056	2767872
425	478	16	128	6144	12288	2611200	2936832
479	538	15	120	5760	11520	2759040	3098880
539	608	14	112	5376	10752	2897664	3268608
609	688	13	104	4992	9984	3040128	3434496
689	782	12	96	4608	9216	3174912	3603456
783	892	11	88	4224	8448	3307392	3767808
893	1024	10	80	3840	7680	3429120	3932160
1025	1186	9	72	3456	6912	3542400	4098816
1187	1388	8	64	3072	6144	3646464	4263936
1389	1648	7	56	2688	5376	3733632	4429824
1649	1996	6	48	2304	4608	3799296	4598784
1997	2480	5	40	1920	3840	3834240	4761600
2481	3210	4	32	1536	3072	3810816	4930560
3211	4424	3	24	1152	2304	3699072	5096448
4425	6852	2	16	768	1536	3398400	5262336
6853	14136	1	8	384	768	2631552	5428224

The number of records that can be recorded on a track depends on the record size. The following equations can be used to determine the number of equal length records per track.

For 2305 Model 1:

$$N = \frac{14,568}{432 + K + KL + DL}$$

Where: K = 202 if KL ≠ 0  
K = 0 if KL = 0

For 2305 Model 2:

$$N = \frac{14,858}{198 + K + KL + DL}$$

Where: K = 91 if KL ≠ 0  
K = 0 if KL = 0

*Note: When an end-of-file mark is written, the data length in the count area must be zero. The storage control, however, adds a one-byte data area when writing the end-of-file mark. Programmers working with track balance routines must allow for this byte by subtracting one byte from the track balance remaining. The standard 432 byte overhead allowance should, therefore, be increased to 433 for each end-of-file record written. Due to differences in recording methods, the 2305 Model 2 overhead allowance (198) must be increased to 200 for each end-of-file record written.*

2305 Model 1 Capacity							
With Keys							
Bytes Per Record ( $K_L + D_L$ )		Records Per				Byte Capacity Per Module	
Minimum	Maximum	Track	Cylinder	Module	Facility	Minimum	Maximum
29	58	21	168	8064	16128	233856	467712
59	94	20	160	7680	15360	453120	721920
95	132	19	152	7296	14592	693120	963072
133	174	18	144	6912	13824	919296	1202688
175	222	17	136	6528	13056	1142400	1449216
223	276	16	128	6144	12288	1370112	1695744
277	336	15	120	5760	11520	1595520	1935360
337	406	14	112	5376	10752	1811712	2182656
407	486	13	104	4992	9984	2031744	2426112
487	580	12	96	4608	9216	2244096	2672640
581	690	11	88	4224	8448	2454144	2914560
691	822	10	80	3840	7680	2653440	3156480
823	984	9	72	3456	6912	2844288	3400704
985	1186	8	64	3072	6144	3025920	3643392
1187	1446	7	56	2688	5376	3190656	3886848
1447	1794	6	48	2304	4608	3333888	4133376
1795	2287	5	40	1920	3840	3446400	4373760
2279	3008	4	32	1536	3072	3500544	4620288
3009	4222	3	24	1152	2304	3466368	4863744
4223	6650	2	16	768	1536	3243264	5107200
6651	13934	1	8	384	768	2553984	5350656

2305 Model 2 Capacity							
With Keys							
Bytes Per Record ( $K_L + D_L$ )		Records Per				Byte Capacity Per Module	
Minimum	Maximum	Track	Cylinder	Module	Facility	Minimum	Maximum
21	27	47	376	36096	72192	758016	974592
28	34	46	368	35328	70656	989184	1201152
35	41	45	360	34560	69120	1209600	1416960
42	48	44	352	33792	67584	1419264	1622016
49	56	43	344	33024	66048	1618176	1849344
57	64	42	336	32256	64512	1838592	2064384
65	73	41	328	31488	62976	2046720	2298624
74	82	40	320	39720	61440	2273280	2519040
83	91	39	312	29952	59904	2486016	2725632
92	102	38	304	29184	58368	2684928	2976768
103	112	37	296	28416	56832	2926848	3182592
113	123	36	288	27648	55296	3124224	3400704
124	135	35	280	26880	53760	3333120	3628800
136	148	34	272	26112	52224	3551232	3864576
149	161	33	264	25344	50688	3776256	4080384
162	175	32	256	24576	49152	3981312	4300800
176	190	31	248	23808	47616	4190208	4523520
191	206	30	240	23040	46080	4400640	4746240
207	223	29	232	22272	44544	4610304	4966656
224	241	28	224	21504	43008	4816896	5182464
242	261	27	216	20736	41472	5018112	5412096
262	282	26	208	19968	39936	5231616	5630976
283	305	25	200	19200	38400	5433600	5856000
306	330	24	192	18432	36864	5640192	6082560
331	357	23	184	17664	35328	5846784	6306048
358	386	22	176	16896	33792	6048768	6521856
387	418	21	168	16128	32256	6241536	6741504
419	453	20	160	15360	30720	6435840	6958080
454	493	19	152	14592	29184	6624768	7193856
494	536	18	144	13824	27648	6829056	7409664
537	585	17	136	13056	26112	7011072	7637760
586	639	16	128	12288	24576	7200768	7852032
640	701	15	120	11520	23040	7372800	8075520
702	772	14	112	10752	21504	7547904	8300544
773	853	13	104	9984	19968	7717632	8516352
854	949	12	96	9216	18432	7870464	8745984
950	1061	11	88	8448	16896	8025600	8963328
1062	1196	10	80	7680	15360	8156160	9185280
1197	1361	9	72	6912	13824	8273664	9407232
1362	1568	8	64	6144	12288	8368128	9633792
1569	1833	7	56	5376	10752	8434944	9854208
1834	2187	6	48	4608	9216	8451072	10077696
2188	2682	5	40	3840	7680	8401920	10298880
2683	3425	4	32	3072	6144	8242176	10521600
3426	4663	3	24	2304	4608	7893504	10743552
4664	7140	2	16	1536	3072	7163904	10967040
7141	14569	1	8	768	1536	5484288	11188992

2305 Model 2 Capacity							
Without Keys							
Bytes Per Record		Records Per				Byte Capacity Per Module	
Minimum	Maximum	Track	Cylinder	Module	Facility	Minimum	Maximum
21	23	67	536	51456	102912	1080576	1183488
24	27	66	528	50688	101376	1216512	1368576
28	30	65	520	49920	99840	1397760	1497600
31	34	64	512	49152	98304	1523712	1671168
35	37	63	504	48384	96768	1693440	1790208
38	41	62	496	47616	95232	1809408	1952256
42	45	61	488	46848	93696	1967616	2108160
46	49	60	480	46080	92160	2119680	2257920
50	53	59	472	45312	90624	2265600	2401536
54	58	58	464	44544	89088	2405376	2583552
59	62	57	456	43776	87552	2582784	2714112
63	67	56	448	43008	86016	2709504	2881536
68	72	55	440	42240	84480	2872320	3041280
73	77	54	432	41472	82944	3027456	3193344
78	82	53	424	40704	81408	3174912	3337728
83	87	52	416	39936	79872	3314688	3474432
88	93	51	408	39168	78336	3446784	3642624
94	99	50	400	38400	76800	3609600	3801600
100	105	49	392	37632	75264	3763200	3951360
106	111	48	384	36864	73728	3907584	4091904
112	118	47	376	36096	72192	4042752	4259328
119	125	46	368	35328	70656	4204032	4416000
126	132	45	360	34560	69120	4354560	4561920
133	139	44	352	33792	67584	4494336	4697088
140	147	43	344	33024	66048	4623360	4854528
148	155	42	336	32256	64512	4773888	4999680
156	164	41	328	31488	62976	4912128	5164032
165	173	40	320	30720	61440	5068800	5314560
174	182	39	312	29952	59904	5211648	5451264
183	193	38	304	29184	58368	5340672	5632512
194	203	37	296	28416	56832	5512704	5768448
204	214	36	288	27648	55296	5640192	5916672
215	226	35	280	26880	53760	5779200	6074880
227	239	34	272	26112	52224	5927424	6240768
240	252	33	264	25344	50688	6082560	6386688
253	266	32	256	24576	49152	6217728	6537216
267	281	31	248	23808	47616	6356736	6690048
282	297	30	240	23040	46080	6497280	6842880
298	314	29	232	22272	44544	6637056	6993408
315	332	28	224	21504	43008	6773760	7139328
333	352	27	216	20736	41472	6905088	7299072
353	373	26	208	19968	39936	7048704	7448064
374	396	25	200	19200	38400	7180800	7603200
397	421	24	192	18432	36864	7317504	7759872
422	448	23	184	17664	35328	7454208	7913472
449	477	22	176	16896	33792	7586304	8059392
478	509	21	168	16128	32256	7709184	8209152
510	544	20	160	15360	30720	7833600	8355840
545	584	19	152	14592	29184	7952640	8521728
585	627	18	144	13824	27648	8087040	8667648
628	676	17	136	13056	26112	8199168	8825856
677	730	16	128	12288	24576	8318976	8970240
731	792	15	120	11520	23040	8421120	9123840
793	863	14	112	10752	21504	8526336	9278976
864	944	13	104	9984	19968	8626176	9424896
945	1040	12	96	9216	18432	8709120	9584640
1041	1152	11	88	8448	16896	8794368	9732096
1153	1287	10	80	7680	15360	8855040	9884160
1288	1452	9	72	6912	13824	8902656	10036224
1453	1659	8	64	6144	12288	8927232	10192896
1660	1924	7	56	5376	10752	8924160	10343424
1925	2278	6	48	4608	9216	8870400	10497024
2279	2773	5	40	3840	7680	8751360	10648320
2774	3516	4	32	3072	6144	8521728	10801152
3517	4754	3	24	2304	4608	8103168	10953216
4755	7231	2	16	1536	3072	7303680	11106816
7232	14660	1	8	768	1536	5554176	11258880

## Appendix B: Sense Bytes

Sense Byte 0	
Bit 0 Command Reject	<ol style="list-style-type: none"> <li>Invalid command code.</li> <li>Invalid command sequence.</li> <li>Invalid or incomplete argument transferred by a control command.</li> <li>Write portion of file mask violated.</li> <li>Write command truncated while operating on an overflow record (Byte 1-bit 7, operation incomplete also set.)</li> </ol>
Bit 1 Intervention Required	<ol style="list-style-type: none"> <li>Addressed device not physically attached to system.</li> <li>Addressed device not ready.</li> </ol>
Bit 2 Bus Out Parity	<ol style="list-style-type: none"> <li>The 2835 has detected bad parity in data transferred from the channel.</li> </ol>
Bit 3 Equipment Check	<ol style="list-style-type: none"> <li>An unusual hardware condition detected in the channel, storage control, or drive. (Condition further defined in sense bytes 16 through 23.)</li> </ol>
Bit 4 Data Check  See Note 1	<ol style="list-style-type: none"> <li>A data error detected in information received from a disk drive. Byte 2 - bit 1, indicates if error is permanent or correctable.</li> <li>Synchronization of read electronics to drive data was unsuccessful.</li> </ol>
Bit 5 Overrun  See Note 2	<ol style="list-style-type: none"> <li>The storage control received a byte from a drive before the last byte read was accepted by the channel.</li> <li>A data byte was received too late from the channel during a write operation.</li> <li>A command was received too late to be executed.</li> </ol>
Bit 6	<ol style="list-style-type: none"> <li>Not used - set to 0.</li> </ol>
Bit 7	<ol style="list-style-type: none"> <li>Not used - set to 0.</li> </ol>
<p><b>Note 1:</b> <i>Byte 2 - bit 1 on indicates correctable error; bytes 18 through 22 contain error correction information as shown in Format 2. Byte 7 indicates specific nature of the condition.</i></p> <p><b>Note 2:</b> <i>Detection of an overrun immediately stops data transmission. When writing, the remaining portion of the record area is padded out with the last byte received.</i></p> <p><i>If the overrun condition exists after retry is exhausted, byte 1 - bit 0 (permanent error) is posted with overrun. Without byte 1 - bit 0, it indicates that command retry was inhibited.</i></p> <p><i>Sense byte 7 distinguishes between data overrun and command overrun.</i></p>	

Sense Byte 1	
Bit 0 Permanent Error	<ol style="list-style-type: none"> <li>Storage control retry has been attempted and was unsuccessful.</li> </ol>
Bit 1 Invalid Track Format	<ol style="list-style-type: none"> <li>An attempt has been made to write data exceeding track capacity.</li> <li>An attempt to read data that has been written past index.</li> </ol>
Bit 2 End of Cylinder	<ol style="list-style-type: none"> <li>A multi-track read or search operation has attempted to continue beyond the addressable limit of the pseudo-cylinder boundary.</li> <li>An overflow operation has attempted to continue beyond the addressable limit of the pseudo-cylinder boundary. (Byte 1 - bit 7, operation incomplete also set.)</li> </ol>
Bit 3	<ol style="list-style-type: none"> <li>Not used - set to 0.</li> </ol>
Bit 4 No Record Found  Note 1	<ol style="list-style-type: none"> <li>Two index points sensed in the same command chain without an intervening read operation in a data area.</li> <li>Two index points sensed in the same command chain without an intervening write, sense, or control command.</li> </ol>
Bit 5 File Protected	<ol style="list-style-type: none"> <li>A seek command has violated the file mask.</li> <li>A multi-track read or search operation has violated the file mask.</li> <li>An overflow operation has violated the seek portion of the file mask. (Byte 1 - bit 7, operation incomplete also set.)</li> </ol>
Bit 6	<ol style="list-style-type: none"> <li>Not used - set to 0.</li> </ol>
Bit 7 Operation Incomplete	<ol style="list-style-type: none"> <li>One of the following conditions occurred during the processing of an overflow record: <ol style="list-style-type: none"> <li>Overflow to a file protected boundary. (Byte 1 - bit 5, file protected also set.)</li> <li>Overflow past the pseudo-cylinder boundary. (Byte 1 - bit 2, end of cylinder also set.)</li> <li>A data error was detected.</li> <li>A write truncation occurred. (Byte 0 - bit 0, command reject also set.)</li> </ol> </li> </ol>
<p><b>Note 1:</b> <i>See the Search Operations section for special consideration about logging no record found in oriented and unoriented search modes.</i></p>	

Sense Byte 2	
Bit 0 Buffered Log Full	1. The buffered log is 75% full and a read buffered log command should be issued to obtain the outstanding log information.
Bit 1 Correctable See Note 1	1. Indicates that the data check posted in sense byte 0 - bit 4, is correctable. Sense bytes 1B through 22 identify the error pattern and error pattern displacement.
Bit 2 through 7	1. Not used - set to 0.
Sense Byte 3	
Bits 0 through 7 Restart Command	<p>1. When byte 1 - bit 7 (operation incomplete) is set, this byte identifies the operation in progress when the interrupt occurred.</p> <p>0000 0110 = A read operation was in progress.</p> <p>0000 0101 = A write operation was in progress.</p> <p>When byte 1 - bit 7 is zero, sense byte three is zero.</p>
Sense Byte 4	
Bits 0 through 7	1. Not used - set to 0.
Sense Byte 5	
Bits 0 through 7 Cylinder Address	1. Identifies the cylinder address of the most recent seek argument from the channel.
Sense Byte 6	
Bits 0 through 7 Head Address	1. Identifies head address of last seek. Head address is updated during multi-track and overflow operations.
<b>Note 1:</b> With byte 0 - bit 4 (data check) bytes 1B through 22 contain error correction information as shown in Format 2.	

Sense Byte 7 - Message Code	
Bits 0 through 7 (Hex)	
04	File mask violation - multitrack
05	File mask violation - Seek
07	File mask violation - overflow
0B	No record found - Search
09	Index before Address Mark - Space Content
0C	No record found - Read
10	PCI Fetch Interrupt
20	End of Cylinder - MT
21	End of Cylinder - O'flow Rec.
30	Data Check - count area
31	Data Check - key area
34	Data Check - data area (Model 2)
35	Data Check - data area (Model 1)
3B	Data Check - No R0 Found
3D	Data Check - Sync Byte Missing
3E	Data Check - AM Missing on Retry
3F	Data Check - Overskew (Model 1)
40	Index detected during Space Count
42	Read/Write past index
58	Write truncation on overflow record.
80	Invalid command
82	Invalid prerequisite
83	Invalid data length
84	Invalid argument transferred
85	Second alternate track spare attempted
8A	Write Mask Violation
8B	Diagnostic Write - Mask Violation
90	Invalid retry command
91 Note 1	Equipment Check - CU or Drive
92	Retry Aborted
94	Index missing
95	APC Failure
9F	Invalid Error Branch
A0	Bus out parity
C0	File inoperative
C1	Inline contention
D0	Command overrun
D1	Data overrun
E1	MPL File Read Check
E2	MPL File Seek Check
E4	MPL File Not Ready
F0	Buffered log full
<b>Note 1:</b> With byte 0, bit 3 (equipment check), bytes 16 through 23 contain storage control and module check indicators as shown in Format 1.	



Sense Byte 8 - Cylinder (1)																												
Bits 0 through 7	High order cylinder byte from ID area of last record processed.																											
Sense Byte 9 - Cylinder (2)																												
Bits 0 through 7	Low order cylinder byte from ID area of last record processed.																											
Sense Byte 10 - Head (1)																												
Bits 0 through 7	High order head byte from ID area of last record processed.																											
Sense Byte 11 - Head (2)																												
Bits 0 through 7	Low order head byte from ID area of last record processed.																											
Sense Byte 12 - Record																												
Bit 0 through 7	Record number from ID area of last record processed.																											
Sense Byte 13 - Sector																												
Bit 0 through 7	Sector number of last record processed.																											
Sense Byte 14																												
	<table><tr><th></th><th>Model 1</th><th>Model 2</th></tr><tr><td>Bit 0</td><td>Switched to channel B.</td><td>Switched to channel B.</td></tr><tr><td>Bit 1</td><td>Diagnostic mode.</td><td>Drive in diagnostic mode.</td></tr><tr><td>Bit 2</td><td>Spare sel - path 0.</td><td>Spare head selected.</td></tr><tr><td>Bit 3</td><td>Spare sel - path 1.</td><td>Not used - set to 0.</td></tr><tr><td>Bit 4</td><td>Side B selected.</td><td>Not used - set to 0.</td></tr><tr><td>Bit 5</td><td>Not used - set to 0.</td><td>Not used - set to 0.</td></tr><tr><td>Bit 6</td><td>Path 0.</td><td>Not used - set to 0.</td></tr><tr><td>Bit 7</td><td>Path 1.</td><td>Not used - set to 0.</td></tr></table>		Model 1	Model 2	Bit 0	Switched to channel B.	Switched to channel B.	Bit 1	Diagnostic mode.	Drive in diagnostic mode.	Bit 2	Spare sel - path 0.	Spare head selected.	Bit 3	Spare sel - path 1.	Not used - set to 0.	Bit 4	Side B selected.	Not used - set to 0.	Bit 5	Not used - set to 0.	Not used - set to 0.	Bit 6	Path 0.	Not used - set to 0.	Bit 7	Path 1.	Not used - set to 0.
	Model 1	Model 2																										
Bit 0	Switched to channel B.	Switched to channel B.																										
Bit 1	Diagnostic mode.	Drive in diagnostic mode.																										
Bit 2	Spare sel - path 0.	Spare head selected.																										
Bit 3	Spare sel - path 1.	Not used - set to 0.																										
Bit 4	Side B selected.	Not used - set to 0.																										
Bit 5	Not used - set to 0.	Not used - set to 0.																										
Bit 6	Path 0.	Not used - set to 0.																										
Bit 7	Path 1.	Not used - set to 0.																										
Sense Byte 15																												
Bit 0 through 7	Not used - set to 0.																											

Sense Bytes 16 - 19				
Sense Byte	Bit	Format 1 - Equipment Check Model 1	Format 1 - Equipment Check Model 2	Format 2 Correctable Data Check** Model 1 and 2
16	0	Overskew*	SERDES Parity Overrun	Not used
	1	Data Overrun	IR Reg Parity	
	2	Mark Out Check	CBO Reg Parity	
	3	Fetch Counter Chk	ECC Check 1	
	4	ECC Check 1	ECC Check 2	
	5	ECC Check 2	Not used	
	6	ECC Input Check	Byte Counter Check	
17	7	Byte Counter Check		Not used
	0	Ser/Des Parity 0	IW Reg Parity	
	1	Ser/Des Parity 1	Drive/Bit Ring Check	
	2	IW Register Parity 0	CUEND*	
	3	IW Register Parity 1	PLO Pulse Missing	
	4	IR Register Parity 0	VFO Phase Error	
	5	IR Register Parity 1	Channel Check	
18	6	SKBO Check 0	Data Error*	Displacement 1
	7	SKBO Check 1	CUDI Check	
	0	Ser/Des Ring 0	Drive Select Check	
	1	Ser/Des Ring 1	CUDI Invalid Tag	
	2	VFO Phase Error	Device Check	
	3	Missing PLO Pulses	TA Reg Check	
	4	ECC Decoder Check	CUDI Reg Check	
19	5	Channel Check	TD Reg Check	Displacement 2
	6	Data Error*	Search Compare Chk	
	7	CU/DI Check	ECC Check	
	0	Drive Selection Chk	Not used - set to 0	
	1	CU/DI Invalid Tag	Not used - set to 0	
	2	Device Check	Not used - set to 0	
	3	TA Register Check	Not used - set to 0	
	4	CU/DI Register Chk	Not used - set to 0	
	5	TD Register Check	Not used - set to 0	
	6	Search Compare Chk	Not used - set to 0	
	7	Not used - set to 0	Not used - set to 0	

\* Does not cause equipment check.

\*\* Refer to error correction functions.

Sense Bytes 20 - 23				
Sense Byte	Bit	Format 1 - Equipment Check Model 1	Format 1 - Equipment Check Model 2	Format 2 Correctable Data Check Model 1 and 2
20	0	Inoperative	Inoperative	Pattern 1
	1	Disk Speed	Disk Speed	
	2	APC Failure	APC Failure	
	3	APC Sync	APC Sync	
	4	378 Jump	378 Jump	
	5	378 Sequence	378 Sequence	
	6	378 Illegal	378 Illegal	
	7	PLO Sync	PLO Sync	Pattern 2
	0	Bus Out Check	Bus Out Check	
	1	Not used - set to 0	Not used - set to 0	
	2	Not used - set to 0	Not used - set to 0	
	3	Bus In Check	Bus In Check	
	4	Multipath Check	Not used - set to 0	
	5	Not used - set to 0	Not used - set to 0	
21	6	Not used - set to 0	Not used - set to 0	
	7	Path 1 Check	Not used - set to 0	
22	0	Clip Check	Clip Check	Pattern 3
	1	Address Reg Check	Address Reg Check	
	2	Write Past Index	Not used - set to 0	
	3	Read Seq Check	Read Seq Check	
	4	Not used - set to 0	Not used - set to 0	
	5	Write Seq Check	Write Seq Check	
	6	Simultaneous Rd/Wr	Simultaneous Rd/Wr	
	7	Current Sink On	Current Sink On	
23	0	Write Transition Chk	Write Transition Chk	Not used
	1	Not used - set to 0	Not used - set to 0	
	2	Write Driver Check	Write Driver Check	
	3	Current Source Chk	Current Source Chk	
	4	High Current Source	High Current Source	
	5	Slider Select	Slider Select	
	6	Read Bias	Read Bias	
	7	Not used - set to 0	Not used - set to 0	

## Appendix C: Log Entry Format

Log Entry																		
Entry Type	Byte 0				Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7							
	0	1	2	3								7						
Control Unit Selective Reset	0	0	1		Storage Error Register	TC Register	TG Register	Failing Address		Error Latches								
								High	Low	0 - 7	8 - 15							
MPL File Check	0	1	1		0    3    4    7													
					Data Check Count	Seek Check Count												
Usage/Overrun Counters	1	0	0	4 D r v 1	6 O F f l o w	Overrun Counters		O F f l o w	A.M. Check Count	0 O F f l o w	Usage Counter							
						Chan A	Chan B				0 - 7	8 - 15	16 - 23	24 - 31				
Correctable Data Check	1	1	0	Retry Count	Command Byte	Device Address	0	1	2	3	4	5	6	7	Cylinder Address	Head Address	Record Number	Message Code
							C h a n	D i s p a r e	S p a r e	S p a t h	P a t h	B 0	1					
Uncorrectable Data Check	1	1	1	Retry Count	Command Byte	Device Address	0	1	2	3	4	5	6	7	Cylinder Address	Head Address	Record Number	Message Code
							C h a n	D i s p a r e	S p a r e	S p a t h	P a t h	B 0	1					

\* Model 1 Only

# Index

## A

Address Marker Area 4  
Alternate Track Spare 3, 47

## B

Branching in Channel Programs 9  
Buffered Log 68

## C

Capabilities 1  
Capacity, Table of  
    Model 1 76  
    Model 2 78  
Channel Command Word 7, 12  
Channel Commands 10  
    Control Commands 10  
    Figure of 13  
    Read Commands 10  
    Search Commands 10  
    Update Write Commands 10  
    Write Commands 10  
Channel Operation (*see*, Input/Output Operations) 7  
Channel Programs  
    Example 1 54  
    Example 2 57  
    Example 3 59  
Channel Selection Switch 71  
Characteristics  
    Data 4  
    General 1  
Command Chaining 8  
Command Descriptions 13  
Command Retry 64  
Contingent Connection 8  
Control Commands 13  
Correction Code Check 6  
Count Area 4

## D

Data Area 4  
Data Chaining 8  
Data Errors  
    Correctable 73  
    Sense Bytes, Figure of 80  
Data Integrity 6  
Data Length 4  
Data Records 5  
Defective Track 3, 47  
Device Release 44  
Device Reserve 45, 71  
Device Status 8, 72  
Diagnostic Load 25  
Diagnostic Write 26  
Diagnostics 68  
Disconnect In 68

## E

Enable/Disable 69, 70  
End-of-File 63, 64

Ending Status 7  
Erase 49  
Error Correction 4, 73  
Error Correction Code 6  
Error Correction Function 73  
    Error Condition Table 73, 74  
    Example for Model 2 73  
    Recovery Action Table 74, 75  
    Sense Bytes, Figure of 80  
Error Detection and Corrections  
    Data Integrity  
        Correction Code Check 6  
        Full Read-Back Check 6  
    Error Correction Code 6  
    Parity 6

## F

Features (*see also*, Special Features)  
    Command Retry 64  
    End-of-File 63  
    Formatting Overflow Records 61  
    Multiple Requesting 64  
    Multiple Track (MT) Operation 61  
    Record Overflow 61  
File Protection 67  
Fixed Head Storage Module (*see*, 2305 Fixed Head Storage) 1  
Flag Byte 4  
Format, Track (*see*, Track Format) 5  
Format, Record (*see*, Record Format) 4  
Full Read-Back Check 6

## G

Gaps 5

## H

Home Address 5

## I

Identifier 4  
Initial Status 10  
Input/Output Operations 7  
    Branching in Channel Programs 9  
    Channel Operation 7  
        Channel Command Word 7  
        Status Presentation 7  
    Command Chaining 8  
    Data Chaining 8  
    Description 7  
    Unit Selection and Device Addressing 9  
Instantaneous Connections 71

## K

Key Area 4  
Key Length 4

## L

Log Entry, Figure of 84  
Long Connections 71

## M

Mask, Set File 21  
Microcodes, Special 68  
Multiple Requesting 64  
Multiple Track (MT) Operation 61

## N

NO-OP 14

## O

OLTEP 68  
Operator Controls and Indicators 69, 70  
Orient 15  
Overflow Record  
    Formatting 63  
    Processing 62

## P

Parity 6  
Pending Status 7  
Power Off 69  
Power On 69

## R

Read Buffered Log 43  
Read Commands 13  
Read Count 35  
Read Count, Key and Data 39  
Read Data 37  
Read Diagnostic Status 46  
Read Home Address 34  
Read Initial Program Load 40  
Read Key and Data 38  
Read R0 36  
Read Sector 41  
Ready Indicator 70  
Recalibrate 16  
Record Format  
    Address Marker Area 4  
    Count Area  
        Data Length 4  
        Error Correction 4  
        Flag 4  
        Identifier 4  
        Key Area 4  
        Key Length 4  
    Data Area 4  
Record Overflow 63  
Record Zero 5  
Recover Action Table 75  
Resets 72  
Restore 23  
Rotation Position Sensing 65, 66

## S

Search Commands 13  
Search Home Address Equal 27  
Search ID Equal 28  
Search ID Equal or High 30  
Search ID High 29  
Search Key Equal 31  
Search Key Equal or High 33  
Search Key High 32  
Search Operations 61  
Seek 17  
Seek Cylinder 18  
Seek Head 19  
Segments 62  
Sense 42  
Sense Bytes, Figure of 80  
Set File Mask 21  
Set Sector 22, 66  
Space Count 20  
Spare, Alternate Track 3, 47  
Special Features 71  
    Two Channel Switch 71  
Status  
    Ending 7  
    Initial 7  
    Multi-Tagged 72  
    Pending 7  
Status Presentation 7  
Storage Control (*see*, 2835 Storage Control) 1  
System Reset 72

## T

Test I/O 11  
TIC Command 11  
Track Format  
    Data Records 5  
    Figure of 5  
    Gaps 5  
    Record Zero 5  
Track Layout 66  
Two Channel Switch 71

## U

Unit Selection and Device Addressing 9  
Update Write Commands 10  
Usage Meter 69, 70  
Usage/Overrun/Error Counters 68

## V

Vary Sensing 24

## W

Write Commands 10  
Write Count, Key and Data 50  
Write Data 52  
Write Home Address 47  
Write Key and Data 53  
Write R0 48  
Write Special Count, Key and Data 51

## **Numeric**

### **2305 Fixed Head Storage**

Capacity, Table of 76

Figure of 2

Model 1

Description 1

Operation 3

Speed and Capacity 3

Model 2

Description 3

Operation 3

Speed and Capacity 3

Standard Features 61

### **2835 Storage Control**

Diagnostics 68

Figure of 2

Model 1 1

Model 2 1

Operator Controls and Indicators 69

Standard Features 61

Reference Manual for IBM 2835 Storage Control and  
IBM 2305 Fixed Head Storage Module

Publication Number  
GA26-1589-5

READER'S  
COMMENT  
FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems.

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

*Note: Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity    Accuracy    Completeness    Organization    Coding    Retrieval    Legibility

If you wish a reply, give your name and mailing address:

---

---

---

What is your occupation ? \_\_\_\_\_

Number of latest Newsletter associated with this publication: \_\_\_\_\_

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Note: Staples can cause problems with automated mail sorting equipment.  
Please use pressure sensitive or other gummed tape to seal this form.

Cut Along This Line

## Reader's Comment Form

Reference Manual for IBM 2835 Storage Control and IBM 2305 Fixed Head Storage Module (File No. S/360-07, S/370-07, 4300-07) Printed in U.S.A. GA26-1589-5

Fold and Tape

Please Do Not Staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS

PERMIT NO. 40

ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation  
Department G26  
555 Bailey Avenue  
San Jose, California 95150

Fold and Tape

Please Do Not Staple

Fold and Tape



International Business Machines Corporation  
Data Processing Division  
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation  
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation  
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601



Publication Number  
GA26-1589-5

File Number  
S/360-07, S/370-07, 4300-07

Printed in  
USA

**IBM**®